Navigation conjointe dans une base de vidéos et d'images

Ibrahima Mbaye

Université Mohammed V Souissi de Rabat

ENSIAS

ÉCOLE NATIONALE SUPÉRIEURE

D'INFORMATIQUE ET D'ANALYSE DES SYSTÈMES

UFR : Réseaux et Télécoms (R&T)

Année 2006

Navigation conjointe dans une base de vidéos et d'images

THÈSE

pour obtenir le titre de

DOCTEUR EN SCIENCES APPLIQUÉES

Discipline : INFORMATIQUE

Présentée et soutenue publiquement par

Ibrahima MBAYE

le 14 Novembre 2006 à 17h

à l'École Nationale Supérieure d'Informatique et d'Analyse des Systèmes (ENSIAS) de Rabat

devant le jury composé de :

Président	:	Driss ABOUTAJDINE, Professeur	FS, Rabat
Rapporteur	:	Abdelkrim BEKKHOUCHA, Professeur	FST, Mohammadia
Rapporteur	:	Azedine BOULMAKOUL, Professeur	FST, Mohammadia
Rapporteur	:	Mohammed EL-KOUTBI, Professeur	ENSIAS, Rabat
Examinateur	:	Mohammed ERRADI, Professeur	ENSIAS, Rabat
Directeur de thèse	:	Rachid OULAD HAJ THAMI, Professeur	ENSIAS, Rabat
Co-directeur de thèse	:	José MARTINEZ, Professeur	Université de Nantes

Laboratoires : SI2M, équipe WiM, ENSIAS - Rabat - Maroc
LINA (Laboratoire d'Informatique de Nantes Atlantique) - France

NAVIGATION CONJOINTE DANS UNE BASE DE VIDÉOS ET D'IMAGES

Joint Navigation in videos and images database

Ibrahima MBAYE

ENSIAS

LABORATOIRE D'INFORMATIQUE DE NANTES ATLANTIQUE

Université Mohammed V Souissi de Rabat

Ibrahima MBAYE

Navigation conjointe dans une base de vidéos et d'images

xxiv+146 p.

Résumé

Dans cette thèse, nous avons développé $Find^{Me}_{DIA}$ destiné à répondre aux besoins de conservation du patrimoine culturel marocain filmé et photographié.

Nous avons donc à gérer une base d'images et de vidéos. Une vidéo pouvant être perçue comme une succession d'images fixes, nous avons traité les vidéos comme une extension qui s'appuie sur la modélisation des images de manière quasi transparente. Notre principal but est, d'une part, de répondre aux besoins de *généricité* et de *flexibilité* permettant de traiter ces différents types de médias visuels et, d'autre part, de proposer un système qui permette de naviguer en basculant indistinctement entre images et vidéos.

Pour la modélisation des vidéos, nous avons proposé $Find^{Vi}_{DEO}$ [66]. En partie modèle, en partie métamodèle, $Find^{Vi}_{DEO}$ est flexible et englobe une large gamme d'applications et de modèles préexistants.

Pour la navigation, nous appliquons la technique des treillis de Galois sur une base de données composée d'images clés extraites des vidéos ainsi que d'images fixes. Le système $Find^{Me}_{DIA}$ [68, 69] résultant est générique et offre la possibilité d'utiliser plusieurs techniques de descriptions des images en vue de la navigation.

Pour tester l'intérêt de nos approches, la modélisation des images clés (extraites des vidéos) et des images fixes est effectuée par $Click^{Im}_{AGE}$ [64] qui propose une représentation semi-structurée des données basée sur le contenu des images.

Mots-clés : vidéo, image, indexation, modélisation, navigation, treillis de Galois

Abstract

In this work, we developed $Find^{Me}_{DIA}$, the purpose of which is to preserve the Moroccan cultural heritage in the form of movies and photographs.

Therefore, we has to manage an joint image and video database. Since a video can be perceived as a sequence of fixed images, we have been treating a video as an extension which depends on the modelling of images in a quasi-transparent way. Our main objective was, on the one hand, to meet the genericity and flexibility needs allowing to navigate with different types of visual data, and, on the other hand, to put forward a system which enables us to navigate by moving indistinctly between images and videos.

As far as the modelling of video is concerned, we proposed $Find^{Vi}_{DEO}$ [66]. Both in its model and metamodel parts, $Find^{Vi}_{DEO}$ is flexible and includes a large spectrum of applications and pre-existing models.

For the sake of navigation, we reused a Galois' lattice technique on a database composed of still images and key-frames extracted from videos. The resulting $Find^{Me}_{DIA}$ system [68, 69] is generic and enables us to use many image description techniques for navigation.

To test the interest of these approaches, the modelling of key-frames (extracted from videos) as well as still images is carried out by $Click^{Im}_{AGE}$ [64] which proposes a semi-structured representation of data based on the content of images.

Keywords: video, image, indexation, modelling, navigation, Galois' lattice

Remerciements

Je voudrais exprimer ma plus grande reconnaissance à M. José Martinez, Professeur à l'école polytechnique de l'université de Nantes qui n'a ménagé aucun effort pour le bon déroulement de cette thèse. Par son implication totale, son dévouement, ses idées et ses conseils, il a été le garant de la réussite de cette thèse.

J'exprime ma plus sincère gratitude à M. Rachid Oulad Haj Thami, Professeur à l'ENSIAS, qui a su diriger cette thèse avec rigueur et patience. Votre disponibilité et vos qualités humaines ont suscité en moi une grande admiration.

Je remercie le Professeur et l'académicien Driss Aboutajdine, Professeur à la Faculté des Sciences de Rabat Agdal, pour son aide et pour l'honneur qu'il me fait en présidant mon jury de thèse.

Je suis profondément reconnaissant à M. Mohammed El-Koutbi, Professeur à l'ENSIAS, M. Abdelkrim Bekkhoucha et M. Azedine Boulmakoul, Professeurs à la Faculté des Sciences et Techniques de Mohammadia, pour avoir acceptés d'évaluer le manuscrit de ma thèse en tant que rapporteurs et de faire parti de mon jury de thèse. Je les remercie pour leurs remarques et conseils qui m'ont été d'un grand apport.

Je tiens aussi à remercier M. Mohammed Erradi, Professeur à l'ENSIAS, d'avoir accepté de juger ce travail et de faire parti de mon jury de thèse en tant qu'examinateur.

Je remercie mes camarades de l'équipe WiM du laboratoire SI2M de l'ENSIAS qui ont partagé avec moi les moments de doutes et les moments d'espoir.

Lors de mes séjours réguliers à Nantes, j'ai eu le plaisir de découvrir un laboratoire agréable et accueillant. Je remercie notamment les membres de l'équipe Atlas-GRIM pour m'avoir permis de m'intégrer rapidement.

Je tiens à remercier pour leurs corrections et conseils tous ceux qui ont relu ce manuscrit.

Enfin, je remercie tous les gens que je n'ai pas cités, amis, famille, ou chercheurs, et qui ont contribué, d'une façon ou d'une autre, à la réussite de cette « aventure ».

Dédicace

Je dédie ce travail à mes parents. Aucun mot n'est assez fort pour leur exprimer la reconnaissance sincère que je leur porte pour la richesse de leurs enseignements. Merci du fond du cœur.

Table des matières

Partie II — Proposition

Liste des tableaux

Partie I — État de l'art

Partie II — Proposition

Table des figures

Partie I — État de l'art

Partie II — Proposition

CHAPITRE 1

INTRODUCTION GÉNÉRALE

Cette thèse s'est déroulée dans le cadre du projet de coopération franco-marocaine sur les STIC (Système des Techniques d'Information et de Communication) de l'INRIA (Institut National de Recherche en Informatique et en Automatique). L'approche que nous avons suivie au cours de ce travail a été de proposer un modèle en vue de répondre à la problématique de la recherche d'informations visuelles. Notre base de test concerne le patrimoine culturel marocain. Après avoir évoqué cette problématique, nous donnerons dans cette introduction un aperçu des résultats obtenus. Nous présenterons ensuite la structure de la thèse afin d'en guider sa lecture.

1.1 Problématique et contexte

Les médias visuels numérisés (images et vidéos) exigent de grandes capacités de stockage et une importante puissance de traitement. Avec l'émergence des normes de compressions, des réseaux à large bande et des ordinateurs personnels de grande puissance, ces médias sont devenus de plus en plus présents, y compris auprès du grand public. Cela amène sinon la nécessité du moins l'opportunité de développer des systèmes de gestion d'information multimédia offrant des outils adaptés quoique analogues à ceux offerts par les systèmes de gestion de bases de données pour les données alphanumériques structurées et les systèmes de recherche d'information pour les données textuelles semi-structurées. L'objectif est de permettre (i) aux concepteurs de modéliser les données multimédias, d'extraire des informations sur leur contenu et de les stocker, et (ii) aux utilisateurs de manipuler et de rechercher par leur contenu l'information *visuelle*.

Malheureusement, à la différence des données textuelles, les images fixes et la vidéo se présentent sous une forme brute n'offrant aucune sémantique. Il est difficile pour le concepteur d'un système de recherche multimédia d'extraire des informations pertinentes et plus

encore d'en déduire des informations sémantiques. Pour la même raison, il est difficile pour un utilisateur de décrire objectivement l'information qu'il cherche.

Dans nos travaux de thèse, nous nous sommes intéressés à ces deux problèmes à savoir l'indexation (description) et la recherche des médias visuels.

L'indexation de données visuelles fait l'objet de nombreux travaux de recherche [64, 113, 26]. C'est une fonction essentielle de tout système de recherche d'information : c'est l'étape pendant laquelle un document se voit conférer un statut conceptuel. On peut distinguer deux phases complémentaires dans le processus d'indexation : la structuration et l'annotation.

La phase de structuration est pour partie automatisable et s'appuie sur les techniques d'analyse du signal [9]. Pour les images, elle permet de mettre en évidence des régions caractéristiques de l'image, entre autres. Dans le cas de la vidéo, la structuration peut consister à retrouver les segments vidéo issus de la phase de montage, voire la structure narrative temporelle dans le cas idéal. Ces informations sont traduites par une structure, respectivement spatiale et temporelle, ainsi que diverses métadonnées de bas niveau décrivant le contenu des images et des vidéos.

La phase d'annotation consiste alors à associer des métadonnées aux différents segments et images. Malheureusement, l'extraction automatique d'une quelconque information sémantique est extrêmement difficile, et même impossible en l'absence totale de connaissances du domaine d'application [1].

L'annotation manuelle est alors l'approche généralement suivie pour l'association d'information sémantique pertinente aux images et aux vidéos. Toutefois, elle souffre d'au moins deux défauts. D'une part, elle est longue et fastidieuse pour des bases de millions d'images ou de centaines d'heures de vidéos. D'autre part, elle introduit la subjectivité de « l'indexeur » quant à la description du contenu des images.

Il est donc essentiel d'avoir des techniques permettant une annotation automatique ou tout du moins semi-automatique pour faciliter la description d'une information pertinente dans une masse d'images et de vidéos, tout en permettant l'ajout de propriétés structurées lorsque l'application ressent le besoin ou offre les moyens de l'associer aux métadonnées décrivant le contenu.

Une fois l'information visuelle indexée, le processus de recherche doit offrir à l'utilisateur des moyens d'exprimer son besoin selon un modèle de requête plus ou moins souple, plus ou moins caché à l'utilisateur. Deux formes d'interrogation interactive sont possibles.

[1] Par exemple, Petković et Jonker [85] proposent un système de règles pour identifier les objets et les événements contenus dans une vidéo. Pour un match de football : « si la forme d'une région est ronde, sa couleur blanche et que cet objet se déplace, alors cet objet est un ballon ».

La première approche consiste à prendre en compte l'utilisateur dans une boucle de rétroaction. Les documents répondant à une requête, ordonnés suivant leur degré de pertinence décroissant, subissent un jugement de pertinence de la part de l'utilisateur. À partir de cette information, le système infère une nouvelle requête, plus proche du besoin effectif de l'utilisateur, qui donne lieu à une nouvelle évaluation. La boucle se poursuit jusqu'à ce que l'utilisateur soit satisfait ou qu'il décide que la base ne contient pas l'information recherchée.

La deuxième approche consiste à pré-traiter la base avec un algorithme de classification afin de la structurer. Si le résultat est un graphe, et pas seulement un ensemble de clusters, alors l'utilisateur pourra naviguer sur ce dernier. L'étape de jugement de pertinence est simplifiée puisqu'au lieu de fournir une liste d'exemples et de contre-exemples, l'utilisateur n'a qu'à suivre un arc du graphe. De la même façon, il n'y a plus de résolution effective d'une requête puisque l'ensemble de ces dernières a été pré-calculé lors de la phase d'indexation. Bien sûr, il faut alors que la structure du graphe permette de traduire le plus de requêtes possibles.

Le présent manuscrit présente notre contribution dans ce domaine de l'indexation et de la recherche d'information visuelle. Notre terrain d'application est, comme nous l'avons déjà évoqué, la conservation du patrimoine culturel marocain filmé et photographié. Les types de vidéos sur lesquels nous travaillons sont des documentaires. Néanmoins, certains films pourraient être indexés, voire des messages publicitaires de l'office du tourisme. Les images fixes seront également utilisées, permettant de mieux appréhender les objets artisanaux, par exemple, ces dernières étant de meilleure qualité visuelle et de plus grande taille que les vidéos. Nous avons donc à gérer un Système de Gestion de Base de données (SGBD) de vidéos et d'images. D'un point de vue applicatif, notre but est de produire un système qui permette de naviguer en basculant indistinctement entre images et vidéos, et ce en s'appuyant sur les ressemblances visuelles.

Pour atteindre nos objectifs, nous nous sommes appuyés sur un treillis de Galois, une structure de graphe permettant d'organiser les éléments d'une relation binaire. Une telle structure de navigation a plusieurs avantages vis-à-vis d'une approche classique basée sur des requêtes formelles. En particulier, il permet d'affranchir l'utilisateur d'une phase de rédaction de requêtes.

Une vidéo pouvant être perçue comme une succession d'images fixes, nous proposons de modéliser ces dernières puis de traiter les vidéos comme une extension qui s'appuie sur la modélisation des images de manière quasi transparente. Les informations temporelles contenues dans la vidéo sont aussi exploiter car une vidéo illustre un scénario, elle possède

un langage avec sa logique et sa grammaire ainsi que des techniques précises de montage qu'il faut utiliser.

L'interrogation se fera donc non pas sur la structure interne des documents mais à partir de la nature visuelle des médias. Il est plus facile de reconnaître quelque chose d'intéressant plutôt que de le décrire. La *navigation* par le contenu semble donc indispensable dans ce contexte. (Naviguer sur des bases de données structurées a déjà été largement exploré par ailleurs [41, 101].) En effet, la navigation est une technique issue des travaux sur les hypertextes qui a démontré, lorsqu'elle s'appuie sur une organisation pertinente des données, que la recherche dans une base était même plus aisée et efficace qu'en effectuant des requêtes [78]. Par ailleurs, le système de navigation sur des images fixes, $Click^{Im}_{AGE}$ [64], est au cœur de notre travail. En fait, un système manipulant conjointement des images *et* des vidéos est une extension assez naturelle, bien que demandant un travail conséquent. La base d'images fournit des illustrations de grande qualité visuelle tandis que les vidéos ont un rôle d'animation.

1.2 Résultats obtenus

Les systèmes d'indexation multimédia peuvent être classés en deux types. Il existe d'une part les systèmes génériques permettant l'indexation de médias se rapportant à différents domaines sans prendre en compte des informations de nature contextuelle. D'autre part, les systèmes spécifiques ne permettent d'indexer que des médias d'un domaine bien particulier (images journalistiques, imagerie médicale, vidéo de surveillance, matchs de football, etc.). Dans ce cas, l'indexation est contextuelle car basée sur une problématique précise. L'utilisation d'un tel système est limitée au domaine d'application.

Notre objectif étant l'indexation des vidéos et images du patrimoine culturel marocain, nécessairement variés, les outils que nous proposons doivent être les plus génériques possibles.

Comme nous l'avons précisé, le développement d'un système complet de recherche d'information multimédia passe par deux étapes principales à savoir l'indexation et la recherche.

Pour l'indexation, une vidéo pouvant être vue comme une succession d'images liées entre elles par des relations temporelles, nous avons modélisé ces dernières et ensuite nous avons proposé une modélisation des vidéos qui s'appuie sur cette modélisation de manière quasi transparente.

Comme la base de données du patrimoine culturel marocain est constituée de vidéos et

d'images, il n'est pas possible d'intégrer systématiquement les premières comme niveau de décomposition le plus fin des secondes. Le système va donc être composé d'un SGBD de vidéos et d'images.

En fait, le système de modélisation des images ($Click^{Im}_{AGE}$) a déjà été développé par notre équipe de l'université de Nantes [64]. $Click^{Im}_{AGE}$ est un système de navigation dans une base de données d'images. Pour $Click^{Im}_{AGE}$, la représentation des données est un ensemble de métriques basées sur le contenu et la forme des images.

Notre travail s'est principalement axé sur la vidéo puis sur les mécanismes de liaison entre le système de navigation sur les vidéos que nous avons développé et le système $Click^{Im}_{AGE}$.

Ainsi, s'inspirant de différents systèmes existant, nous avons proposé un modèle assez générique et flexible [66, 67, 69] capable de prendre en compte la diversité des vidéos du patrimoine culturel marocain. Il peut être vu à la fois comme un guide dans l'élaboration d'un modèle particulier et comme un modèle suffisamment générique pour pouvoir supporter un grand nombre de métadonnées associées à des vidéos variables. Cette flexibilité se traduit, d'une part, par une décomposition hiérarchique paramétrable des types de vidéos et, d'autre part, par la variété des descripteurs que l'on peut associer à chaque décomposition d'une vidéo.

Les éléments indivisibles de la vidéo, c'est-à-dire les images, seront alors décrits et indexés *via* les mécanismes mis en œuvre par $Click^{Im}_{AGE}$.

Si le modèle répond aux besoins de généricité et de flexibilité, nous nous appuyons sur la navigation pour répondre aux besoins de performances et de facilité de recherche des documents archivés. Nous avons proposé une technique de navigation dans des bases de vidéos se basant sur la navigation sur les treillis de Galois tel qu'implémenté dans $Click^{Im}_{AGE}$ avec notamment des modifications pour l'adapter à la navigation sur les vidéos. Le système $Find^{Vi}_{DEO}$ [68] résultant est un système de navigation dans une base de données vidéos.

Nous appelons la liaison entre les deux sous-systèmes $Find^{Me}_{DIA}$. $Find^{Me}_{DIA}$ permet de naviguer dans une base de vidéos et d'images en basculant indistinctement entre images et vidéos.

L'utilisation d'un système générique permet l'utilisation de plusieurs techniques d'indexation et de recherche. Ainsi, les descriptions retenues peuvent variées d'une application à une autre. Elles peuvent être liées au contenu intrinsèque des images comme la couleur, ou au contraire peuvent ajouter des sémantiques au contenu à travers des mots clés.

Les vidéos et les images ont été modélisées avec UML (*Unified Modeling Language*) mais nous avons choisi une base de données relationnelle pour le stockage. Leur maturité,

leur large adoption dans l'industrie, ainsi que leur tenue en charge, en adéquation avec la quantité très importante de données à traiter, en font aujourd'hui le meilleur candidat pour l'implémentation. Toutefois, la généricité du modèle rend possible une implémentation différente, par exemple au-dessus d'un SGBD (Système de Gestion de Base de Données) XML (*eXtensible Markup Language*) ou à objets.

Les tests de performances effectués montrent que la demande initiale de tenue en charge est respectée.

1.3 Plan de la thèse

Ce document décrit les différents aspects de nos travaux concernant l'indexation des médias visuels et la navigation dans une base de données vidéos et d'images. Il est divisé en deux parties essentielles :

- La partie I est consacrée à l'état de l'art des modèles d'indexation et de recherche. Elle est composée de deux chapitres :
  - Les propositions existantes pour la modélisation du contenu des médias visuels sont très variées. Par conséquent, le chapitre 2 introduit une vue synthétique suivant plusieurs points de vue. Nous décrirons d'abord les éléments constitutifs d'une base de données d'images. En quelques mots, la modélisation d'une image au travers d'un ensemble de propriétés de natures très diverses (couleur, texture, format...). Nous passons ensuite en revue quelques familles de modèles d'indexation de la vidéo (« rudimentaires », applicatifs, visuels et génériques) en faisant ressortir les apports importants. Pour faciliter la comparaison, les schémas des propositions de la littérature ont tous été traduits dans un même formalisme, en l'occurrence UML.
  - Dans le chapitre 3 nous passons en revue quelques techniques de recherche dans des bases de données visuelles en mettant davantage l'accent sur la navigation.
- La partie II de ce mémoire présente nos principales propositions. Elle comprend quatre chapitres :
  - Au chapitre 4, nous rappelons d'abord les résultats obtenus pour $Click^{Im}_{AGE}$ pour la modélisation des images. Ensuite nous présentons l'utilisation des treillis de Galois par $Click^{Im}_{AGE}$ pour la navigation dans une base d'images.
  - Le chapitre 5 présente notre modèle $Find^{Vi}_{DEO}$ pour la modélisation des vidéos, et détaille ses différentes facettes et les motivations de nos choix. Il distingue une structure hiérarchique et les métadonnées à proprement parler. La liaison entre

la structure et les métadonnées est décrite. La liaison avec la base d'images de $Click^{Im}_{AGE}$ est également évoquée. Les caractéristiques du prototype $Find^{Me}_{DIA}$ issue de cette liaison sont aussi présentées.

- Au chapitre 6, nous détaillons comment adapter les techniques de navigation de $Click^{Im}_{AGE}$ pour définir une méthode de navigation dans une base de données vidéos dans un premier temps et par la suite la navigation conjointe dans une base d'images et de vidéos.
- Le chapitre 7 concerne l'implémentation de nos différentes propositions. Ce chapitre nous permet de montrer comment l'implémentation choisie répond aux besoins de performances et supporter de la charge.

- Enfin, dans le chapitre 8, nous concluons en récapitulant les contributions de la thèse et en les mettant en perspective.

PARTIE I

État de l'art

La modélisation des données a toujours été au cœur de tout système de gestion de l'information [39]. En conséquence, elle a été étudiée par des communautés d'horizons différents. Si une convergence des concepts de modélisation se dessine avec le temps, tous les aspects du problème n'ont pas conduit à un accord parfait, en partie à cause des besoins applicatifs particuliers à chaque communauté.

Dans cette partie, nous allons passer en revue les principales propositions de modélisation et d'interrogation des médias visuels. Le domaine étant très vaste, il est difficile d'être exhaustif. Nous nous sommes donc tenus à offrir une vue synthétique et résumée des propositions référencées, sous plusieurs points de vue.

CHAPITRE 2

MODÉLISATION DES MÉDIAS VISUELS

Dans ce chapitre, nous nous intéressons à la modélisation et à l'indexation des médias visuels. La séparation entre ces deux techniques de manipulations des médias visuels est assez artificielle puisque les auteurs abordent généralement ces deux aspects de front dans leurs travaux. Néanmoins, les propositions dans chaque partie étant nombreuses, ce choix de présentation offre l'avantage de la clarté. La traduction des requêtes et la mise en correspondance de ces dernières avec les descriptions des documents seront abordées dans le chapitre 3.

L'indexation a pour objectif une meilleure « compréhension » du contenu des documents afin de permettre une manipulation efficace. Une problématique essentielle pour les médias visuels est l'extraction de l'information la plus pertinente possible, sous toutes ses formes mêmes celles qui peuvent sembler *a priori* anecdotiques (distinction photographie / graphique, intérieur / extérieur [109], ville / campagne [118], jour / nuit, etc.), et la caractérisation de cette information par des descripteurs synthétiques et comparables pour la recherche par similarité et/ou la classification. Indépendamment des travaux visant à mieux appréhender le contenu d'une image, fixe ou animée, il s'agit donc de modéliser ces informations pour pouvoir ensuite les rechercher.

Dans la suite, nous présentons un état de l'art des systèmes de modélisation de l'image et de la vidéo en mettant davantage l'accent sur la vidéo.

2.1 Modélisation de l'image

Toutefois, nous nous étendons au préalable sur les techniques qui consistent à modéliser le contenu des images par extraction de caractéristiques de bas niveau, généralement de

nature statistique, représentant son aspect visuel telles que la couleur [108], la texture [110], etc. Notons que des descriptions plus précises nécessitent des techniques d'analyse d'image avancées: segmentation, extraction de contours, etc. [9].

Dans cette section, nous présentons tout d'abord quelques BDI (Base de Données d'Images), ensuite nous décrirons les éléments constitutifs d'une BDI. En quelques mots, la modélisation des images au travers d'un ensemble de propriétés de natures diverses (couleur, texture, forme, format...). Le choix d'un critère de ressemblance pour les propriétés de l'image prises dans leur ensemble, le plus souvent vues sous la forme d'un vecteur ainsi que le choix de techniques d'interrogation assez variées sera abordé dans le chapitre 3, la principale mettant en œuvre des mécanismes d'inférence afin de déterminer les objectifs des utilisateurs. Tout cela conduit à adopter une architecture extrêmement flexible.

2.1.1 Bases de données d'images

Les bases de données d'images forment l'une des briques des bibliothèques numériques. Les recherches menées depuis quelques années ont abouti à de très nombreux prototypes : Amore (*Advanced Multimedia-oriented Retrieval Engine*), BlobWorld, CANDID [48], Chabot / Cypress [79], CORE (*Content Object Retrieval Engine*) [130], $Find^{Im}_{AGE}$ [63], GeneRIC (système Générique de Recherche d'Image par le Contenu) [113], FIRST (*Fuzzy Image Retrieval SysTem*), IDQS (*Image Database Query System*) [128], ImageRover, Jacob, MARS (*Multimedia Analysis and Retrieval System*) [82], MetaSeek, MIR (*Multimedia Indexing and Retrieval*), MMIS (*Multimedia Information System*), MULTOS (*MULTimedia Office Server*) [72], NeTra [60], Picasso [21], PicHunter, PIQImage, PhotoBook [84], QBIC (*Query By Image Content*) [26], RetrievalWare, SQUID (*Shape Queries Using Image Databases*), SurfImage [77], Virage, VisualSEEK [104], WebSEEK [18], WebSeer, Xenomania, etc.

Bien que les BDI ne soient pas encore communément utilisées, elles ont été intégrées dans des produits commerciaux, comme Ultimedia Manager chez IBM (*International Business Machine*. Ces systèmes peuvent former la base d'applications basées sur les images, comme un outil d'aide à l'aménagement d'intérieurs (choix de peintures, papier-peints, tapis, etc., qui s'harmonisent) ou plus simplement la recherche dans l'album de famille numérique.

2.1.2 Généralités sur les systèmes de recherche d'images

Le cadre le plus général est celui des systèmes ouverts, notamment le *World Wide Web*, où les images et les utilisateurs sont les plus divers. D'une part, il n'est pas possible dans ce cadre de profiter de connaissances *a priori* sur le contenu typique des images. D'autres part, l'utilisateur n'est pas un expert et il ne peut donc ni comprendre nombre des caractéristiques associées aux images ni par conséquent les raisons pour lesquelles certaines images sont renvoyées.

Par opposition, nombre de bases sont restreintes à des classes d'images particulières. Les domaines d'application sont nombreux : images journalistiques (presse, télévision, publicité…), images d'art, images architecturales, imagerie médicale, photographie aérienne et/ou satellite, reconnaissance de photos d'identité, météorologie, etc. Le fait de travailler sur des classes restreintes d'images permet de disposer d'informations supplémentaires qu'il faut exploiter.

À cette première différenciation, on peut en ajouter une autre qui devrait avoir tendance à s'effacer. Les systèmes de recherche d'images (SRi [1]) constituent le noyau qu'il convient d'ajouter à un système de gestion de base de données (SGBD) pour aboutir à une véritable BDI. Ainsi, on peut soutenir qu'une BDI est un cas particulier de système de gestion de base de données multimédia (SGBDMM) [76, 65], alors qu'un SRi s'inscrit dans le cadre plus flou des bibliothèques numériques. L'évolution des SGBD devrait à terme permettre d'englober, ou tout au moins d'offrir le support principal des bibliothèques numériques [1].

2.1.2.1 Problématique

La problématique des « SRi par le contenu » est essentiellement de permettre la recherche d'images en se basant sur le contenu.

Définition du contenu : La première difficulté repose bien sur la définition que l'on veut bien donner au terme « contenu ».

La représentation d'une image est semi-structurée [1] puisque l'on peut lui associer les qualificatifs suivants :

- *irrégulière :* plusieurs types d'images coexistent (en niveau de gris, en couleur, avec ou sans palette de couleurs…);
- *incomplète :* toutes les informations sur le contenu et la sémantique n'ont pas nécessairement été extraites ou fournies pour toutes les images (traitements coûteux,

[1] Le « i » minuscule est employé pour éviter la confusion avec l'abréviation SRI correspondant aux « système de recherche d'*information* », ces derniers incluant aussi les SRi.

sources hétérogènes et/ou réparties, qualité insuffisante...), par exemple dans QBIC [26] ;

- *extensible :* de nouvelles propriétés doivent pouvoir être ajoutées afin de tenir compte de catégories d'images particulières ou de nouvelles techniques d'extraction ;
- *applicative :* les images n'ont pas vocation à être manipulées en isolation mais plutôt en liaison avec un schéma les englobant.

En ce domaine, les différents auteurs s'accordent à reconnaître une hiérarchie entre données de bas niveau, ou primitives, et données de haut niveau (logiques, sémantiques), issue des travaux en vision artificielle [9].

Le niveau le plus bas est le contenu *intrinsèque*, directement lié au signal physique et représenté le plus souvent par une matrice de pixels, la représentation canonique.

Des opérations de traitements d'images permettent d'améliorer la qualité des images numérisées en réduisant le bruit, augmentant le contraste, corrigeant certaines aberrations, etc.

Des processus *automatiques* d'analyse d'image sont alors capables d'extraire des informations *quantitatives* et *générales*. Les principaux représentants de cette classe sont les histogrammes et les transformées.

À l'opposé, les données *extrinsèques* et *sémantiques* doivent souvent être fournies manuellement. Le titre, ou le sujet, d'une image fait partie de cette catégorie. Les mots clés constituent la plus grande généralisation possible de métadonnées *extrinsèques* à l'image. Ils peuvent être organisés en *thesauri*. Ces métadonnées sont le plus souvent *non quantifiables*, mais des degrés de croyances peuvent rendre les transitions sur certaines caractéristiques moins abruptes (peu, assez, très...). On rejoint les problèmes de la recherche d'information textuelle : indexation incomplète, ambiguë, variable dans le temps et l'espace [27].

Néanmoins, avec la numérisation croissante des techniques de capture d'images, certains attributs sémantiques peuvent être associés automatiquement à l'image nouvellement créée comme l'auteur, son âge, etc. (récupérés dans un profil de l'utilisateur), la date de prise de vue, voire les coordonnées terrestres, mélange de données sémantiques et quantifiables.

De même, les images associées à des documents offrent des informations connexes. Par exemple, WebSEEK [18], qui indexe les images des pages HTML (*HyperText Markup Language*) utilise le contenu des balises, c'est à dire essentiellement leur URL (*Uniform Ressource Locator*). En définitive, le système profite d'une indexation manuelle qui n'est pas faite au moment de l'insertion dans la base, mais directement « à la source ». Cependant, les catégories sont établies de manière semi-automatique, leur validation étant laissées au soin d'un administrateur.

De plus, il existe des techniques d'automatisation de l'extraction de connaissances sémantiques, applicables nécessairement à des classes réduites d'images :

- après une phase d'apprentissage, des mots clés peuvent être associés automatiquement à des régions d'images (ciel, herbe, feuillage, peau, fourrure...) [96, 87] ;
- dans le domaine des peintures de la Renaissance, Picasso [21] exploite les règles d'un ouvrage d'art afin de dériver jusqu'à des sentiments du contenu de l'utilisation d'une grande quantité de rouge et de bleu;
- Chabot [79] offre un algorithme simple de détection de l'horizon, les images indexées étant des images de paysage;
- Szummer et Picard [109] propose une classification entre les images prises en intérieur et à l'extérieur;
- etc.

Enfin, notons qu'une BDI, contrairement à un système de recherche d'images, offre naturellement des connaissances supplémentaires. Le *schéma de la base de données de l'application* structure le contenu de la base et indirectement celui des images qui y participent. Il permet d'exploiter les associations entre classes d'objets. Par exemple, une image qui est liée à une instance de la classe *Ville* a une probabilité élevée de représenter un paysage urbain.

Bien sûr, le passage entre le contenu *visuel* et les métadonnées est progressif. Par exemple, les techniques de segmentation, permettant d'isoler des régions significatives d'une image, nécessitent déjà quelques hypothèses et peuvent être aussi bien très spécifiques (en imagerie médicale, entre autres) que rester d'un usage assez général (segmentation basée sur la couleur, par exemple).

Pour résumer partiellement, on peut établir une opposition grossière entre les données de bas niveau et celles de haut niveau sur plusieurs critères :

- contenu *intrinsèque à extrinsèque*;
- propriétés *générales à spécifiques*;
- propriétés *quantitatives à qualitatives*;
- extraction *automatique à « manuelle »*;
- propriétés *objectives à subjectives.*

La hiérarchie qui a été définie ci-dessus entraîne une difficulté majeure pour la recherche, notamment quant à la subjectivité. En effet, le système travaille généralement sur des données de bas niveau, tandis que l'utilisateur travaille au niveau le plus haut [100].

Certains auteurs sont des tenants de l'extraction maximale de données sémantiques. En effet, les expérimentations montrent que l'emploi de données de haut niveau, en l'occur-

rence des mots clés, améliore très sensiblement la qualité des résultats [79, 16].

D'autres auteurs défendent une approche où seules les données de bas niveau sont prises en compte explicitement. Il y a deux raisons à l'appui de cette thèse : l'existence d'une sémantique latente et l'introduction de la subjectivité dans la sémantique de haut niveau.

Tout d'abord, la combinaison de données de bas niveau permet d'aboutir à de bons résultats. En effet, il existe une *sémantique latente*. Par exemple, l'herbe est généralement verte, le ciel bleu. Ainsi, CANDID s'appuie-t-il nommément sur cette approche [48], ainsi que FourEyes pour la texture [87], où la combinaison de plusieurs caractéristiques de bas niveau peut faire apparaître des critères de haut niveau *implicite*.

Une autre raison est de limiter volontairement le niveau le plus élevé de connaissance car, devenu *subjectif*, il ne s'applique plus à tous les utilisateurs. Santini et Jain [100] argumentent même sur le fait que, la sémantique d'une image ne pouvant pas être définie formellement, il faut se limiter à chercher des *corrélations* entre les objectifs de l'utilisateur et les caractéristiques qui peuvent être extraites des images. Est-il raisonnable d'exiger du système qu'il retrouve des informations sémantiques d'un niveau extrêmement élevé et absolument absentes de l'image? Par exemple, doit-il être capable de retrouver les images de chefs d'états (de pays arabes) lorsqu'on lui présente un portrait de Sa Majesté Mohammed VI et une image en pied de Ben Ali? Cela est déjà une tâche complexe à partir des seuls noms. Le système de recherche d'images devrait donc se limiter à un système d'aide à la recherche d'images pertinentes, exploitant des capacités liées à la vision mais non une vision automatique. En d'autre termes, le système n'est pas capable de « voir » les images, mais seulement d'aider l'utilisateur à les classer semi-automatiquement.

Définition de la recherche : Une fois que le contenu a été défini, il faut proposer un mécanisme de recherche d'information. Or, les utilisateurs sont divers. Il convient donc d'offrir plusieurs formes d'interrogation. En effet, la représentation de la requête est tout aussi importante que celle des images. Plus précisément, le choix d'un couple de représentation pour les requêtes et les images conditionne le développement d'un système. Le chapitre 3 revient plus longuement sur la définition de la recherche.

2.1.3 Modélisation des images

Nous allons présenter dans cette partie, la modélisation des images comme des objets semi-structurés au travers d'un ensemble de propriétés de natures diverses *a priori*. Plusieurs outils mathématiques ont pu être mis à contribution pour caractériser certains ou plusieurs types de caractéristiques : morphologie mathématique, fractales, statistiques,

transformées (Fourier, ondelettes, Gabor...), etc. Nous mettons davantage l'accent sur les histogrammes qui sont des outils statistiques simples utilisés par $Click^{Im}_{AGE}$ pour la classification des images et par $Find^{Vi}_{DEO}$ pour la segmentation automatique des vidéos en plan.

Les histogrammes ne nécessitent aucun développement mathématique tout en permettant de discuter les aspects importants concernant la détermination de bonnes caractéristiques.

Ensuite, nous montrerons dans le chapitre 3 comment construire des mesures de similarité sur ces propriétés ainsi que sur leur combinaison.

2.1.3.1 Représentation des images

La représentation des images, bien qu'extrêmement importante, n'en reste pas moins très ouverte. Par conséquent, le schéma I d'une classe d'images ne peut être introduit que de manière générique (c'est-à-dire à spécialiser ultérieurement), comme un objet semi-structuré [2] :

$$I = C^{\mathbb{N}} \tag{2.1}$$

où C est un ensemble de polymorphe [3] de caractéristiques. Dans la suite, nous emploierons abusivement la notation C_C pour représenter aussi bien l'ensemble des valeurs de la caractéristique C que la fonction correspondante $I \rightarrow C$.

Trouver des caractéristiques qui permettent de traduire au mieux le contenu d'une image est une tâche difficile. En effet, elles doivent vérifier au maximum les critères suivants, parfois antimoniques [11] :

- *exhaustivité* : les caractéristiques doivent couvrir l'ensemble des éléments (importants) de l'image ;
- *compacité*: le codage de l'information discriminante doit être compact afin de réduire simultanément les coûts de stockage et de traitement, soit *a priori* en choisissant des résumés pertinents, soit *a posteriori* par des techniques de réduction de la dimension [24] ;
- *robustesse* : les caractéristiques doivent pouvoir s'accommoder du bruit résiduel, quasiment incontournable tant les images présentent des défauts divers (prises de vues dans des conditions difficiles, numérisation défectueuse, quantification liée aux images utilisant des palettes de couleurs...) ;

[2] La description sous la forme d'un vecteur est une facilité de notation. La traduction effective devrait être celle d'une fonction qui à un nom de caractéristique associerait sa valeur.

[3] Le sens de polymorphe inclus l'héritage (ou le sous-typage) mais ne se limite pas à cela dans les concepts que nous introduirons à la suite.

- *discrimination* : bien qu'étant amenée à être utilisée de manière complémentaire, chaque caractéristique doit par elle-même permettre de différencier de nombreuses classes d'images ;
- *précision* : autorisant ainsi une discrimination poussée ;
- *automatisation* : les caractéristiques doivent être fournies par des moyens mécaniques afin de réduire les coûts et d'éviter les distorsions.

La condition de compacité exclue de pouvoir utiliser comme caractéristique la représentation canonique de l'image. Par exemple, Chabot [79] ne stocke que des vignettes dans la base de données, les images elles-mêmes étant déportées sur plus de 10 To de disques optiques [79].

Données de bas niveau : Parmi les caractéristiques de bas niveau qui ont été les plus exploitées, car les plus proches des descriptions que peut en faire un être humain, on trouve la *couleur et la texture*.

Plusieurs modèles de couleurs coexistent pour répondre à des besoins différents, depuis les modèles physiques (dont RVB est le principal) jusqu'aux modèles perceptuels (soit les variations autour de la séparation entre teinte, saturation et luminance), en passant par des modèles normalisés (XYZ, L*u*v*, L*a*b*) aux modèles propriétaires (YES, PhotoYCC chez Kodak), présentant ou pas la propriété d'uniformité, jusqu'aux atlas de couleurs dont l'archétype est celui de Munsell (basé sur l'espace HVC).

Quant à la texture, elle est généralement traduite par des caractéristiques de granularité, contraste et direction [26].

Depuis Swaind et Ballard [108], les histogrammes restent à la base d'un très grand nombre de propositions :

$$h : \begin{array}{rl} (E_J) \rightarrow & E \rightarrow [0,1] \\ (e_j)_{j\in J} \mapsto & \left\{ e \mapsto \frac{(e_j | e_j = e)_{j\in J}}{(e_j)_{j\in J}} | e \in (e_j)_{j\in J} \right\} \end{array} \tag{2.2}$$

où $J = X \times Y$ pour les images standard.

(E_J) dénote une famille de données, autorisant des répétitions donc, indicées par des éléments de l'ensemble J afin de différencier les doublons, et $E \rightarrow [0,1]$ est la fonction qui à chaque valeur particulière de la famille de données associe sa fréquence, c'est-à-dire l'histogramme correspondant.

Les histogrammes présentent potentiellement les propriétés d'exhaustivité, de robustesse et de précision. En revanche, ils ne sont ni particulièrement compacts, ni extrêmement discriminants dans de grandes BDI [83].

Le problème de la compacité peut se résoudre plus ou moins aisément en créant des classes de valeurs, permettant de réduire simultanément les temps de traitement. Dans le cas de la couleur, on peut créer des classes perceptuelles, c'est-à-dire un découpage non uniforme de l'espace des couleurs (9 teintes dans $Find^{Im}_{AGE}$ [63]). Ce découpage peut même faire disparaître la tri-dimensionnalité de l'espace, c'est-à-dire remplacer trois histogrammes par un seul mais pour lequel on ne pourra peut être plus faire de mesures statistiques (13 couleurs dans Cypress [15], 27 dans CORE [130], 32 dans [16], 64 dans QBIC [24]).

Le problème de la discrimination est plus profond et général. Il concerne l'absence de corrélation entre les différentes modalités d'un histogramme. Le compactage évoqué dans le paragraphe précédent le résout en partie mais de manière trop brutale puisqu'il limite simultanément le nombre de classes d'images qui peuvent être différenciées.

Si l'utilisation de l'histogramme dans sa totalité n'est pas possible, un nombre suffisant de moments d'inertie (moyenne puis moments centrés, cf. Eq. 2.3) permet de traduire aussi finement qu'on le souhaite l'allure de l'histogramme. La plupart des auteurs se limitent aux deux à quatre premiers moments. Stricker et Orengo [107] montrent que l'approche par moments d'inertie est celle qui combine le maximum d'avantages.

$$\mu \quad : \quad \begin{array}{ll} E \rightarrow & [min_h, max_h] \\ h \mapsto & \sum_{\forall j} j \times h(j) \end{array} \tag{2.3}$$

$$m_n \quad : \quad \begin{array}{ll} E \rightarrow & [min_h, max_h] \\ h \mapsto & \sum_{\forall j} (j - \mu(h))^n \times h(j) \end{array} \tag{2.4}$$

L'interprétation des moments dépend de la caractéristique retenue. Pour un histogramme de niveaux de gris, la moyenne représente tout simplement l'intensité moyenne ; l'écart-type offre une mesure du contraste ; le troisième moment traduit l'asymétrie, c'est-à-dire qu'il corrige fortement les deux premiers moments ; enfin, le moment d'ordre quatre représente l'aplatissement.

Pour la texture, on utilise également les premiers moments de l'histogramme des niveaux de gris, au moins quatre. Toutefois, l'agencement spatial est prépondérant et d'autres techniques doivent être utilisées pour déterminer la granularité et la direction[4] ainsi que d'autres caractéristiques de la texture, comme la rugosité, la périodicité, la régularité, la

[4]Pour la direction, il est possible d'utiliser la technique des histogrammes en calculant le gradient sur chaque pixel. L'histogramme est défini sur l'intervalle discrétisé $[0, 2\pi[$ et fait apparaître les (macro) directions privilégiées de l'image.

complexité, etc. [43]. De plus, il ne semble pas que l'on dispose encore d'un meilleur modèle de représentation de texture [88, 87].

Malgré tout, les histogrammes, ou les moments d'inertie, restent une mesure trop globale de l'image. Dans des bases importantes, plusieurs images tout à fait différentes ont des histogrammes très proches les uns des autres.

Les données de niveau intermédiaire : Les informations extraites jusqu'ici portent essentiellement sur la totalité de l'image. Or, l'information la plus pertinente après la couleur est la disposition spatiale des principales couleurs. Cela fait partie du processus de prévision, inconscient. En effet, ces couleurs correspondent généralement à des objets réels. Sans prétendre à ce niveau de précision, une image peut être décomposée en parties intéressantes, en faisant un minimum d'hypothèses.

Afin de fournir des indications plus précises sur l'arrangement spatial des pixels dans l'image grâce à la technique des histogrammes, on peut tout d'abord faire des hypothèses minimales sur leur composition. Ainsi peut-on considérer que dans la majorité des cas le sujet principal d'une image se trouve proche du centre de l'image [26]. De toute façon l'œil est naturellement attiré par le centre [37]. On peut étendre ce raisonnement au cas des images composées suivant les règles canoniques de l'art photographique en effectuant un découpage suivant les lignes de forces (découpage en tiers, horizontalement et verticalement) [106]. Pour améliorer la précision des informations spatiales, des découpages récursifs de l'image (*quad-trees* [98, 47]) peuvent être effectués jusqu'à une profondeur ou un niveau d'homogénéité fixés. Enfin, des techniques plus évoluées permettent de déterminer et/ou d'exploiter la concentration spatiale des couleurs: auto-corrélogrammes [38], rétro-projection [15], transformation Ragon [125], triangulation de Delaunay [112], etc.

Pour aller plus loin, il faut recourir à des techniques d'analyse d'image. Les deux principaux traitements sont la segmentation et l'extraction de contour [9].

Les techniques de segmentation, très nombreuses, permettent d'extraire des régions de l'image présentant une certaine homogénéité, suivant un critère donné. Des critères courants sont l'homogénéité en niveaux de gris ou en couleur (les études de psychologie cognitives montrant que l'œil est particulièrement sensible au larges zones de couleur homogènes [37, 11]) et à l'homogénéité de texture. Des versions manuelles ou semi-automatisées (remplissage par inondation à partir d'un point désigné par l'utilisateur, contour actif [26]) permettent d'obtenir des régions significatives, munies éventuellement d'informations sémantiques, mais pour un coût élevé.

Tous les calculs d'histogrammes peuvent être étendus aux régions. Mais, les régions

permettent de tirer avantage de nouvelles caractéristiques pour améliorer les résultats des recherches.

À chaque région r on peut associer des caractéristiques additionnelles de forme et d'agencement spatial :

- surface (s_r) ;
- périmètre (p_r) ;
- orientation (θ_r) ;
- position absolue : barycentre ($C_x(r) = \mu(h(r))$ et $C_y(r) = \mu(h(r))$), rectangle minimum d'encadrement (x_r, y_r, l_r, h_r tenant éventuellement compte de θ_r) ;
- position relative : distance euclidienne au centre de l'image ($C_c(r) = L_2((h/2, l/2), (Cx(r), Cy(r)))$) ;
- forme : étirement, excentricité ou élongation (l_r/h_r), rectangularité ($s_r/(l_r \times h_r)$), circularité ou compacité ($4\pi s_r/{p_r}^2$), approximations par transformée de Fourier [12], moments d'inertie [43], angles et vecteurs tangents [26], etc.

De plus, entre régions, on peut associer des caractéristiques binaires, voire n-aires :

- comparaisons sur les différentes propriétés ;
- relations spatiales : avec (1) relations de Allen [5] (before, meets, overlaps, starts, during, finishes et leurs inverses ainsi que equals), (2) coordonnées paramétriques (ρ, θ) [121], ou (3) graphe de relations entre objets de l'image, et même (4) de simples histogrammes bidimensionnels d'adjacence de régions sur leur couleur respective [28].

Les données de haut niveau : Les données de haut niveau sont considérées par certains auteurs comme les plus utiles pour mener à bien des recherches pertinentes dans une base d'images [16]. Le principe de l'indexation de haut niveau consiste à fournir des descriptions sémantiques de la scène et des objets du monde réel qui s'y trouvent. Malheureusement, la sémantique doit généralement être fournie manuellement car elle nécessite une compréhension de la scène visible sur l'image, sauf cas particuliers évoqués précédemment.

L'approche par mots clés [15] est l'alternative la plus générale qui permet de décrire aussi bien objectivement que subjectivement, intrinsèquement qu'extrinsèquement le contenu de l'image :

$$C_M : I \rightarrow 2^{A*} \tag{2.5}$$

Au-delà, les modèles proposés sont difficilement comparables. On peut toutefois distinguer les informations portant sur (1) les régions de l'image et les informations perceptuelles associées, (2) les objets du monde réel et les informations sémantiques associées, et enfin

(3) la mise en correspondance entre (1) et (2). Les points (2) et (3) constituent donc l'apport de ce niveau.

$EMIR^2$ [70] introduit un graphe orienté structurel décrivant la structure composite des objets (maison composée d'un toit et de murs...). Les informations spatiales sont portées par les nœuds (points, segments, polygones...) et les arcs (métriques – close, far –, vectorielles – north, south, east, west – et topologiques – cross, overlap, disjoint, in, touch) d'un graphe de description de la scène en deux dimensions. Enfin, une description symbolique sépare les attributs formatés (auteur, taille...) des concepts génériques. Notons que cette proposition de (trop) haut niveau ne reconnaît pas les objets de l'image ; seuls les objets conceptuels sont décrits ; il n'y a donc pas de correspondance effective. Elle vise des applications où la sémantique domine largement. $EMIR^2$ inclus aussi des notions de pertinence (importance d'un objet) et d'incertitude.

Pour sa part, la proposition de Meghini [71] utilise une segmentation simple où chaque région se voit associer un attribut de couleur. Au niveau de la description sémantique, l'approche à objets est utilisée (classe, héritage et agrégation). La liaison entre les deux niveaux s'établit par une fonction qui à une ou plusieurs régions de l'image associe un objet sémantique. Un langage de requêtes est alors fournit au dessus de cette structure de données.

CORE [130] va plus loin en offrant plusieurs interprétations (composées de concepts) pour une même propriété (composée de mesures) de l'image. Une application intéressante en est le système STAR qui permet des recherches sur des logos d'entreprises. Ceux-ci ne représentent pas toujours un objet réel, mais ils ont une interprétation symbolique.

Ces quelques propositions suffisent à montrer à quel point les approches peuvent varier dans les détails, même si l'on peut y déceler quelques points communs à un niveau d'abstraction très élevé.

2.2 Modélisation de la vidéo

La modélisation d'un document vidéo consiste en l'organisation suivant une structuration claire de son contenu. Contrairement à ce qui se passe pour les données classiques, il faudra tenir compte de ses composantes visuelles et temporelles (et probablement sonores). Les systèmes de recherche de la vidéo par le contenu suivent trois phases distinctes tout au long de leur développement : la structuration, l'annotation et l'organisation en vue de l'exploitation des documents annotés. La première est une étape de segmentation de ces documents, la seconde une annotation des différents segments temporels issus de la phase de segmentation et la dernière étape consiste en l'exploitation (interrogation, navigation, etc.)

des différentes bases de données vidéos constituées.

2.2.1 Structuration et annotation des documents vidéo

Comme pour les images, la représentation d'une vidéo est aussi semi-structurée (cf. section 2.1.2.1).

Plusieurs modèles de description des vidéos ont été proposés. Au-delà de leurs différences, nous essayons de capturer les similitudes, c'est-à-dire les besoins communs à plusieurs approches. Cela nous permettra de proposer au chapitre 5 pour partie un modèle, pour partie un métamodèle fédérateur.

2.2.1.1 Vue d'ensemble

Plusieurs propositions de modélisation de la vidéo ont vu le jour. Nous ne visons pas à l'exhaustivité. De plus, après notre étude, il nous semble difficile de proposer une taxinomie claire. Nous avons plutôt dressé un certain nombre de tableaux. Ils offrent une vue synthétique et résumée des propositions référencées, sous plusieurs points de vue.

Les modèles proposés sont fortement liés aux *applications visées*, d'une part, et aux *techniques d'indexation automatiques*, d'autre part. Les premières amènent souvent à décrire le contenu des vidéos en termes *sémantiques*, tandis que les secondes introduisent des descripteurs de bas-niveau, c'est-à-dire décrivant exclusivement le *contenu*. Les deux approches induisent une *structuration* de la vidéo et génèrent un certain nombre de *métadonnées*. Les *modèles de données* utilisés sont également variables.

Le tableau 2.1 traduit le fait que certaines propositions ont été faites dans un cadre applicatif bien particulier. Les fictions, ou encore les documentaires, s'organisent autour d'un discours narratif bien construit [74, 116]. À l'inverse, les actualités (journaux télévisés [3] mais aussi reportages sportifs [115]) sont plutôt des agrégations de sous-vidéos, sans lien réel bien souvent. La spécialisation à un domaine encore plus fermé (notamment football, tennis, etc.) [129, 86] pemet de s'appuyer sur une connaissance du domaine limitée, aussi bien du point de vue sémantique (noms des coureurs, des écuries, etc.) que pour l'analyse du signal (détection des joueurs sur un court de tennis, poursuite du ballon dans un match de football, extraction des cartouches, etc.). Néanmoins, il est possible d'en extraire des apports d'une portée plus générale.

Le tableau 2.2 regroupe des propositions qui s'appuient essentiellement sur des techniques d'analyse du signal. Elles permettent de structurer de manière simple une vidéo. Les deux opérations fondamentales sont le découpage en plans et l'extraction d'une ou de plu-

Propositions	Applications privilégiées
[74]	fictions, humours
[117]	vidéo-clubs, vidéo à la demande
[3]	journaux télévisés
[129]	football
[86]	formule 1

Table 2.1 – Application(s) privilégiée(s)

Propositions	Analyses du signal vidéo
[103, 74]	détection de plan
[26, 3, 56]	détection de plan extraction d'image caractéristique
[17]	détection de plan extraction d'image caractéristique regroupement en scènes
[85, 114, 7]	suivi de trajectoire d'objet
[115]	détection de plan suivi de trajectoire d'objet

Table 2.2 – Analyse du signal

sieurs images caractéristiques par plan. Certaines essaient de regrouper les plans consécutifs en scènes. Quelques propositions s'appuient sur la norme MPEG-4 (*Motion Pictures Expert Group* et tendent donc à extraire des « objets » visuels en décrivant notamment leur trajectoire [115, 114, 7]. La vidéo est alors décomposée en VO (*visual object*) dont les instances sont décrites par des VOP (*visual objet plane*) correspondant à un suivi de trajectoire.

On peut cependant noter qu'en dehors de cas encore particuliers liés à des manipulations de la vidéo, comme un chanteur placé en incrustation sur des fonds changeants, les objets visuels resteront confinés dans un plan. Ce n'est qu'en exploitant des informations sémantiques (certaines pouvant être déduites par des processus d'apprentissage) que l'on pourra en déduire que certains des objets visuels qui apparaissent sur plusieurs plans correspondent au même objet du « monde réel ». Par exemple, Petkovic et Jonker [85] offrent des règles pour identifier les objets et les événements contenus dans une vidéo. Un exemple de règle pour un match de football peut être formulé comme suit : « si la forme d'une région est ronde, sa couleur blanche et que cet objet se déplace, alors cet objet est un ballon ».

Le tableau 2.3 présente plusieurs propositions qui vont plus loin dans la structuration

Propositions	Hiérarchies de décomposition
[7]	fixe : vidéo / VO / VOP
[103]	fixe : vidéo / plan
[26]	fixe : vidéo / plan / image représentative
[17]	fixe : vidéo / scène / plan / image(s) représentative(s)
[105]	fixe : vidéo / plan / groupe / ligne-de-vie
[36, 114]	fixe : vidéo / séquence / scène / plan
[74]	fixe : film / chapitre / épisode / scène / sous-scène / plan
[3]	variable : journal / information*
[117]	variable : vidéo / * / plan
[36]	variable : [[[* /] séquence /] scène /] plan

Table 2.3 – Décomposition hiérarchique d'une vidéo

Propositions	Types de métadonnées
[103, 36]	strates
[117]	jaquettes, résumés, strates
[3]	mots-clés
[74]	strates, mots-clés
[81, 105, 19]	lignes de temps

Table 2.4 – Types de métadonnées

d'une vidéo que les simples hiérarchies vidéo / plan ou objet / trajectoire issues de l'analyse du signal. La plupart utilisent des hiérarchies de profondeur fixée à l'avance par l'application ou par des considérations généralistes. Bien entendu, une hiérarchisation de taille variable est préférable [36, 117].

Le tableau 2.4 s'intéresse aux types de métadonnées qui sont associées à une vidéo et plus précisément à ses parties. Il ne s'agit là que de quelques possibilités, tant le nombre et le type des métadonnées sont extensibles. Là réside la principale difficulté dans la définition d'un schéma de données et surtout de schémas de données interchangeables entre applications.

Enfin, le tableau 2.5 permet de préciser sur quel modèle de données ont été construites les propositions. Certains modèles sont davantage utilisés pour implémenter une proposition [117] que pour modéliser les données. D'autres sont des spécialisations de modèles

Propositions	Modèles de métadonnées
[117]	UML / relationnel
[71, 62, 32]	logique
[89, 75, 73]	graphes sémantiques, conceptuels

Table 2.5 – Modèle de métadonnées

généraux [62]. D'autres encore forment une application d'un modèle général comme les graphes conceptuels [73]. Soulignons que le choix d'un modèle a des répercussions sur le langage de requête.

2.2.1.2 Apports des propositions antérieures

Ayant maintenant une vue plus générale des différents aspects liés à la modélisation de la vidéo, nous décrivons succinctement quelques propositions, et plus particulièrement leurs apports. Pour cela, nous adoptons une approche qui, du point de vue de la modélisation, va du plus simple au plus générique en passant par les applications spécifiques et les approches visuelles. Pour faciliter la comparaison, les modèles ont été traduits dans un même formalisme, en l'occurrence UML.

Modèles rudimentaires : Les modèles rudimentaires offrent un découpage fixe du contenu des vidéos. Ils sont basés pour la plupart sur des techniques de détection automatique de plans ainsi que sur des techniques de macro-segmentation. Nous reviendrons plus en détail sur ces techniques dans la sous section 2.3

Le modèle introduit par Aguierre et Davenport [103] (cf. figure 2.1) effectue un découpage rudimentaire d'une vidéo en *plans*. Ainsi, dans ce modèle, grâce à l'analyse du signal, une vidéo est transformée d'une suite d'images en une suite de plans. L'information sémantique est introduite sous la forme de *strates*. Une strate peut être perçue comme un mot clé « typé », soit une paire attribut / valeur, telles que objet / bâtiment, bâtiment / école, personnage / coiffeur, animal / chameau, etc. Les strates peuvent être associées à un ou plusieurs plans.

Cette modélisation est simple. Elle s'appuie sur les techniques de base d'analyse automatique du contenu d'une vidéo, à savoir la segmentation en plan. Les strates fournissent pour leur part la sémantique qui n'est pas extraite automatiquement. La description par strates est cependant trop stricte pour analyser complètementt une vidéo. Elle ne permet de décrire ni la dynamique ni la structure du film. Le modèle proposé par Chahir [17] utilise

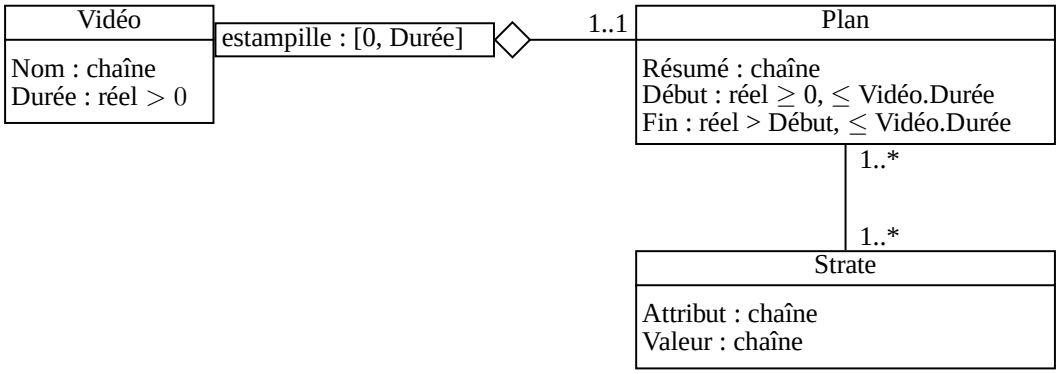

Figure 2.1 – Schéma UML du modèle de Aguierre et Davenport [103]

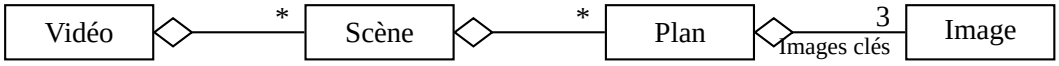

Figure 2.2 – Schéma UML du modèle de Chahir [17]

les résultats des travaux préalablement effectués sur les images fixes pour l'indexation des images animées. Le travail sur les images fixes s'est articulé autour de trois axes : la segmentation, l'extraction d'un résumé visuel et spatial. Chaque vidéo est segmentée en *plans* regroupés en des unités macroscopiques plus sémantiques appelées *clusters* ou *scène.* Un cluster est un ensemble de plan similaire. Ce sont en fait des plans filmés au même endroit et contenant les mêmes objets. Pour minimiser le nombre trop élevé d'images à indexer, les plans ont été représentés par trois images : la première, la dernière et l'image représentative qui est choisie afin de résumer au mieux le contenu visuel du plan (c'est la plus proche d'une image moyenne virtuelle). La constitution de clusters, permet notamment de détecter les plans alternatifs, les plans de coupe et les scènes formées de plans continus. La figure 2.2 montre la modélisation de la vidéo d'après Chahir.

Des modèles similaires à ceux ci ont été proposés. Certains marquent leur différence en proposant une *classification* des métadonnées sous la forme d'une hiérarchie d'héritage [45, 36] : les types des strates correspondent à des classes (cf. les classes d'annotations sur la figure 2.3). L'extensibilité est moins simple, elle dépend du système à objets sous-jacent. D'autres étendent la notion de strate en remplaçant l'unique paire attribut / valeur par des n-uplets de telles paires $[a_1 : v_1, \ldots, a_n : v_n]$ [81].

Modèles applicatifs : Le modèle proposé par Ahanger et Little [3] s'applique au cas particulier des journaux télévisés. Une modélisation UML est présentée sur la figure 2.4. Elle utilise une décomposition assez simple : un journal se décompose en nouvelles. Cette dé-

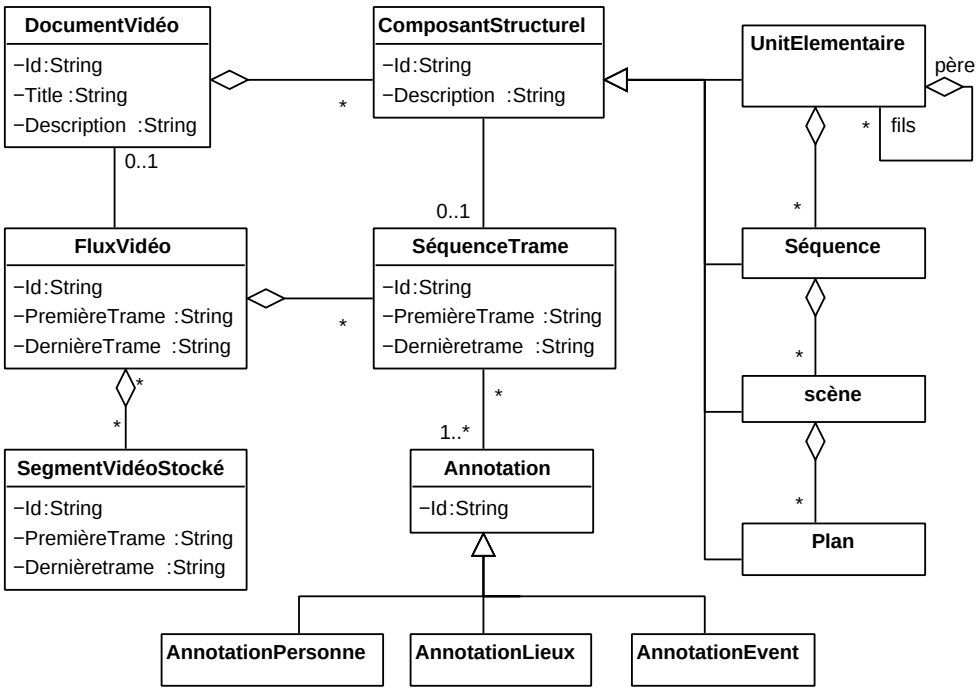

Figure 2.3 – Schéma UML du modèle de Hjelsvold et Midtstraum [36]

composition ne correspond pas à un découpage physique puisque chaque nouvelle constitue une unité sémantique et non physique. De plus, une nouvelle peut être composée d'autres nouvelles et appartenir à une ou plusieurs journaux. En effet, le but est de stocker des archives de journaux télévisés puis de les utiliser en vue de fabriquer des journaux personnalisés. Ces nouvelles sont annotées par des « objets » de différentes natures:

- positions de certains objets visuels ;
- sources des informations ;
- textes (sous-titrage, bandeaux, informations complémentaires) ;
- type d'audio (paroles du présentateur, voix *off*) ;
- type d'intervention (entretien, commentaire, etc.) ;
- partie d'un reportage (introduction, arrière-plan, etc.).

Un objet peut lui-même être composé d'autres objets et appartenir à (ou plutôt référencer) une ou plusieurs nouvelles. Cette liste, exhaustive dans cette proposition, souligne la richesse et la diversité des métadonnées. Typiquement, les approches à objets autorisent l'extensibilité des métadonnées *via* l'héritage [36].

Un modèle plus complexe, développé en vue d'une mise en œuvre dans des vidéo-clubs, est celui de Vachon et Doucet [116, 117]. La figure 2.5 illustre le formalisme UML

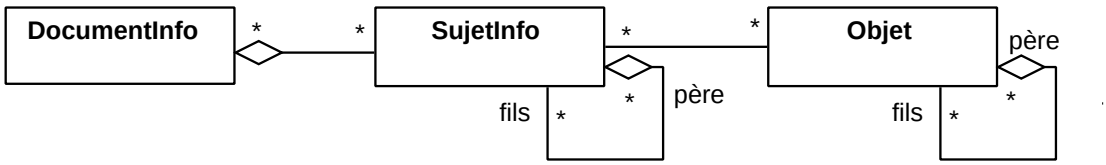

Figure 2.4 – Schéma UML du modèle de Ahanger et Little [3]

de ce modèle. La décomposition hiérarchique de la vidéo est de profondeur *variable*. La racine reste la vidéo, les feuilles restent les plans, mais autant de niveaux intermédiaires que souhaités peuvent être introduits. La seule contrainte est que les instances, c'est-à-dire les arbres associés à une vidéo, soient complets. De la sorte, il existe une implémentation relationnelle des arbres qui permet de rédiger en SQL (*Standard Query Language*) des requêtes « récursives », typiquement déterminer par l'ensemble des plans (feuilles) associés à un nœud interne de l'arbre de description de la vidéo. Les informations sémantiques sont également assez complètes. À une vidéo est associée une jaquette qui comprend des informations telles que le titre, le réalisateur, l'année, les acteurs, les langues, etc. Aux niveaux inférieurs peuvent être associés (i) des strates et (ii) des résumés. Enfin, les plans portent des informations relatives au contenu, à savoir une image représentative mais également une « vidéette » (ou courte miniature vidéo). Notez que l'image représentative ne fait pas partie de la hiérarchie de la vidéo mais constitue une métadonnée. La « vidéette » permet de diffuser sur le réseau, en flux continu, des extraits du plan afin que l'utilisateur puisse se faire une idée assez précise d'un film et procède ensuite à sa location.

Un exemple d'arbre de hauteur 3 est présenté sur la figure 2.6.

Le système AUTEUR [74] découpe la vidéo en Histoire / Épisode / Séquence / Scène / Action / Sous-action. Ce découpage est basé sur les relations temporelles entre la structure de la vidéo et son histoire. La figure 2.7 illustre ce découpage.

Le but du système AUTEUR est d'implémenter un ensemble de méthodes permettant l'édition automatique de la vidéo en se basant essentiellement sur la sémantique de la représentation conceptuelle de la vidéo. La figure 2.8 montre la structure d'une histoire particulière.

Dans cet exemple, le découpage se fait de façon hiérarchique. Une histoire, comme nous le montre la figure 2.8, se décompose en plusieurs épisodes. Un épisode se subdivise en plusieurs séquences, puis en scènes, actions et sous-actions. Le modèle associe aussi la structure d'une histoire avec des strates à chaque feuille.

Modèles exploitant des objets visuels : Plusieurs propositions cherchent à indexer non pas l'artefact que constitue l'image animée, mais plutôt les objets visuels qui apparaissent,

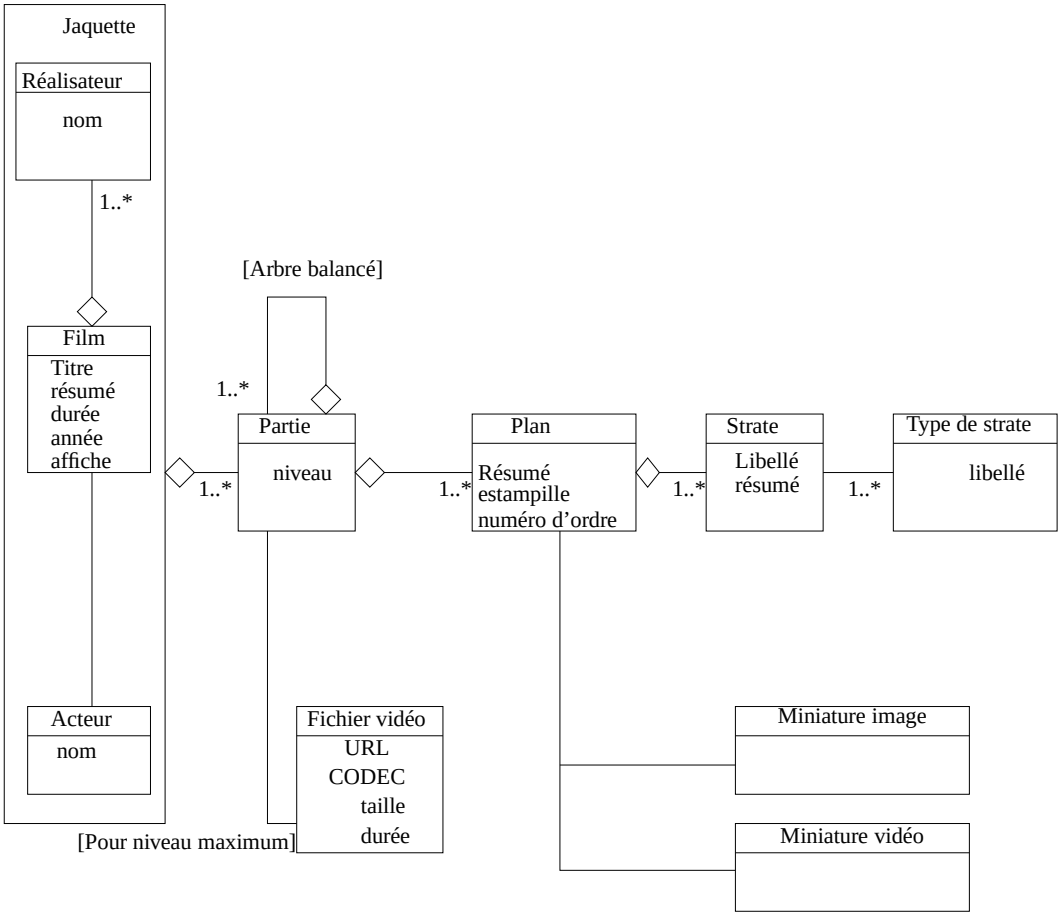

Figure 2.5 – Schéma UML du modèle ARMITAGE [116]

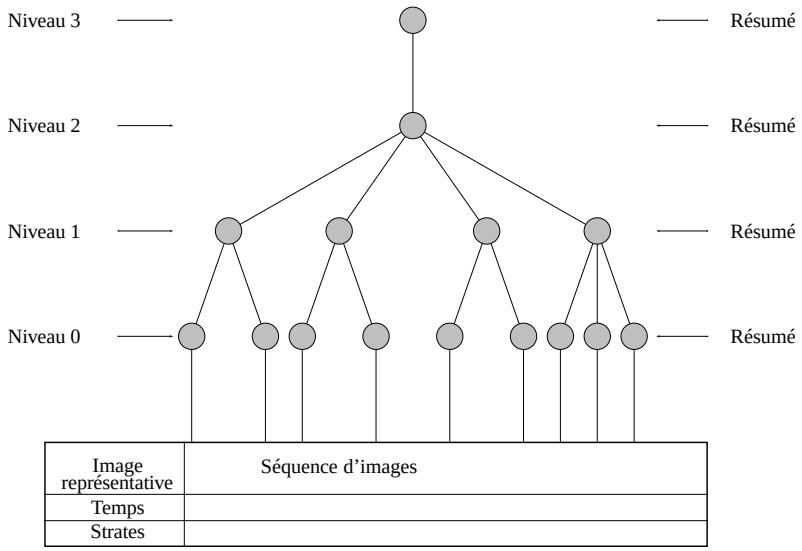

Figure 2.6 – Vision synthétique du modèle ARMITAGE [116]

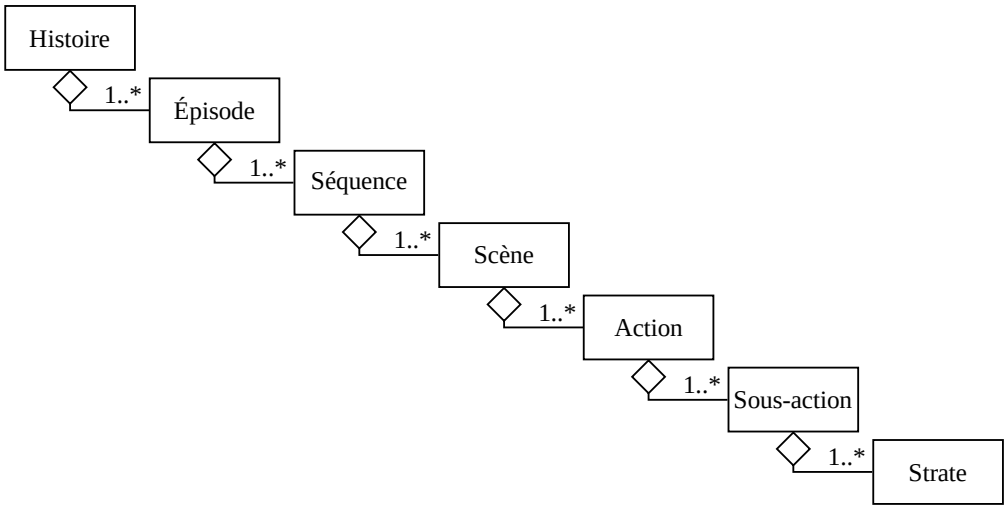

Figure 2.7 – Modélisation du découpage de [74]

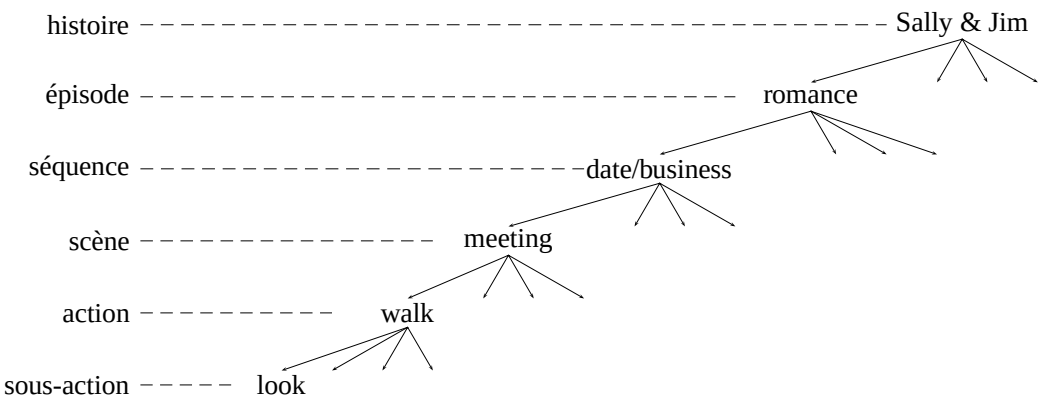

Figure 2.8 – Découpage d'une vidéo, adapté de Nack et Parkes [74]

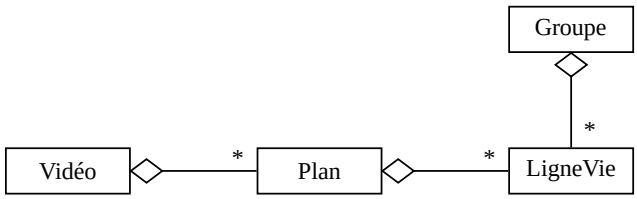

Figure 2.9 – Schéma UML du modèle de Stefanidis *et al.* [105]

se déplacent et disparaissent de l'écran et qui, dans le cas idéal, correspondraient à des objets du « monde réel ».

Sans reconnaissance automatique des objets visuels, la notion de *ligne de vie* d'un objet sémantique a été introduite dans plusieurs propositions [81, 19]. On peut la considérer comme une strate accompagnée d'une information temporelle, un sous-ensemble fini d'intervalles qui précise les périodes pendant lesquelles l'objet est présent sur les images : $(attribut, valeur, \{[t_i, t'_i]\})$.

Stefanidis *et al.* [105] étendent les lignes de vie (*lifelines*) avec la dimension spatiale. Une ligne de vie est définie comme un ensemble de localisations spatiales d'un objet dans un intervalle temporel durant lequel l'objet se déplace: $\{(x, y, t)|t \in [t_1, t_2]\}$ (cf. la figure 2.9). Les évolutions spatio-temporelles sont analysées et les lignes de vie groupées en tenant compte des accélérations et décélérations (changements sur l'axe temporel) et des « rotations » (changements sur le plan).

Ardizzone *et al.* [7] s'appuient directement sur le codage MPEG-4 (Moving Pictures Expert Group) pour extraire les VO (*video object*) et VOP (*video object plane*) correspondant aux trajectoires. La figure 2.10 montre le schéma UML de ce modèle.

Le modèle proposé par Thuong [114] que nous appelons ici modèle MOVI-Opéra car il est issu de la coopération entre les projets MOVI (Modélisation pour la vision par ordi-

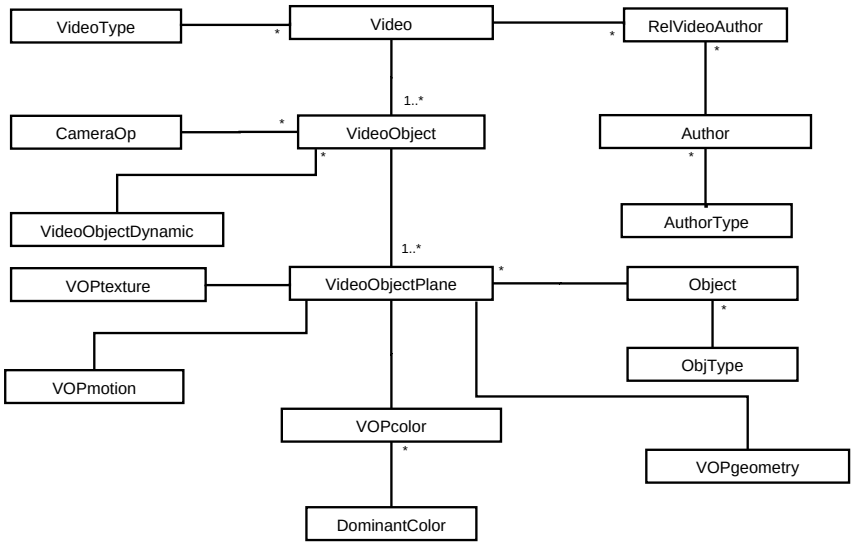

Figure 2.10 – Schéma UML du modèle de Ardizzone *et al.* [7]

nateur) et Opéra (Outils pour les documents électroniques recherche et application) menée à l'INRIA. Il s'agit d'utiliser la structure des vidéos fournie par VideoPrep de MOVI pour l'édition de documents multimédias telle qu'elle est vue dans Opéra. De façon plus large pour tout traitement capable d'utiliser les structures extraites par VideoPrep telle que l'indexation des documents vidéos.

VideoPrep a été développé à l'INRIA dans le cadre du projet MOVI et VISTA (Vision Spatio-Temporelle et Active) en partenariat avec *Alcatel Corporate Research Center* (CRC). C'est un prototype d'environnement qui permet l'extraction de manière semi-automatique des informations de structures d'une vidéo et l'exploration des vidéos ainsi structurées. L'extraction des informations de structures s'effectue en plusieurs étapes : le regroupement des images en plans et l'extraction d'objets contenus dans ces plans. Il associe aussi des descripteurs aux objets extraits (actions, textes, etc.) et forme des classes d'équivalences d'objets (objets sémantiquement équivalents) grâce aux outils de mise en correspondance d'images fixes fournis par MOVI.

La figure 2.11 illustre le découpage de la vidéo d'après [114].

Le schéma proposé est complexe mais intéressant. Dans la même perspective, le modèle VideX [115] combine également une structuration hiérarchique (variable mais limitée à trois niveaux : [[séquence /] scène /] plan) avec le suivi de trajectoire caractérisé par des objets visuels pouvant être composés de plusieurs régions homogènes en couleur et texture et qui subissent des transformations de forme et de déplacements (le schéma UML est fourni dans

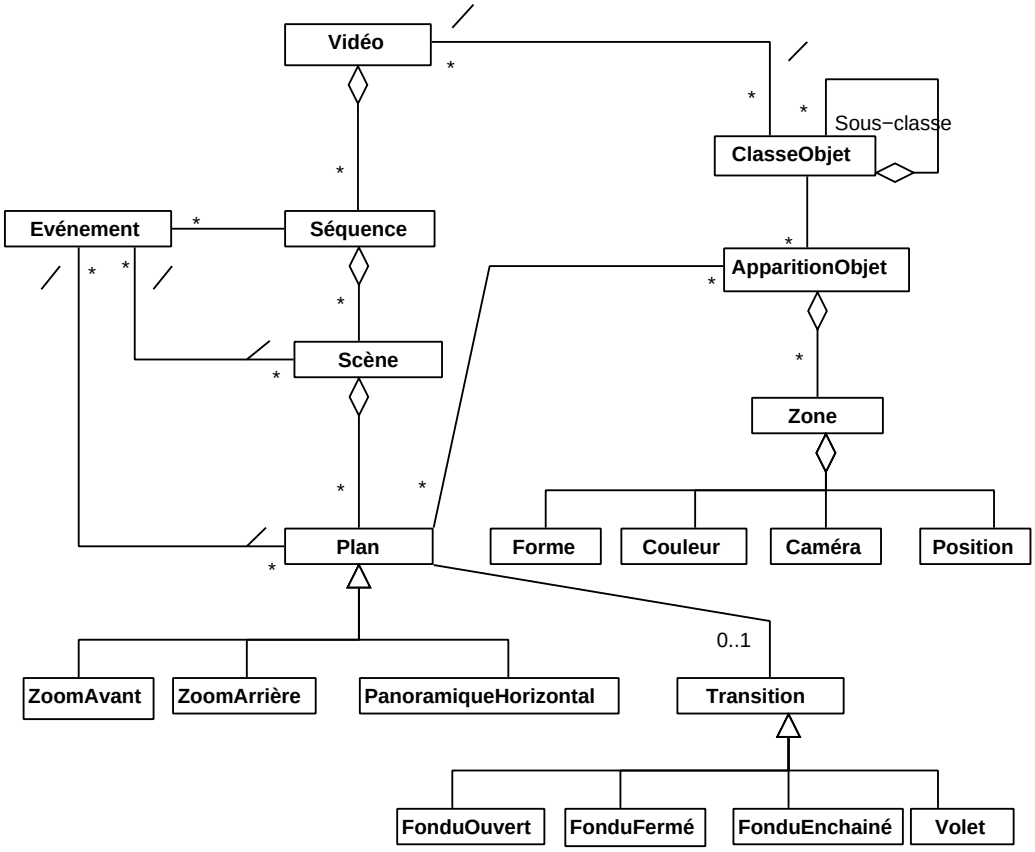

Figure 2.11 – Schéma UML du modèle MOVI-Opéra [114]

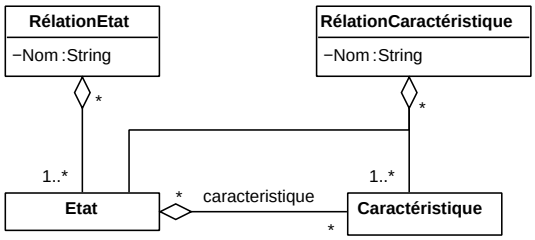

Figure 2.12 – Schéma UML du modèle de Marcus et Subrahmanian [62]

l'article cité). En outre, des annotations peuvent être fournies sous la forme d'une hiérarchie d'héritage (personnes,évènements, lieux et divers).

Modèles génériques : Effectuant un bond important, Marcus et Subrahmanian [62] introduisent un métamodèle extrêmement général dont nous donnons le schéma UML sur la figure 2.12. En fait, il inclut le modèle relationnel, ce qui le rend équivalent à ce dernier en définitive ! Ce modèle propose de décrire un média en distinguant les *états* et les *descripteurs* (*features*).

Les états peuvent correspondre à une décomposition de la vidéo, mais aussi à des éléments apparaissant sur cette dernière, comme des personnages. Les états sont liés entre eux par des relations quelconques. Par exemple, les personnages apparaissant dans une même partie d'une vidéo peuvent être positionnés les uns par rapports aux autres, ce qui pourrait donner `àgauche(Ali, Ibrahim, scène12)`.

En ce qui concerne les descripteurs, des relations quelconques peuvent également être introduites, mais toujours en référence à un état. Néanmoins, pour décrire des faits très généraux, comme « la tour Hassan se situe à Rabat », il est possible d'utiliser des variables libres. Dans l'exemple, cela se traduit par {situerdans(X,"Tour Hassan", "Rabat").}

Le lecteur l'aura compris, il s'agit d'un modèle relationnel [61], et plus précisément logique (Prolog est utilisé [62]), où l'on introduit deux types de données particuliers et où l'on contraint l'ensemble des relations / prédicats :

$$R \subset \bigcup_{i \in N^*} (\acute{E}tat^i) \cup (\acute{E}tat \times Descripteur^i).$$

2.3 Méthodes de structuration des documents vidéo

La structuration des documents vidéo vise à définir des représentations appropriées de leur contenu pouvant être exploitées aisément pour des tâches d'indexation. Il est essentiel de disposer d'outils automatiques, ou tout du moins semi-automatiques, de segmentation de

ces documents (à cause de la taille énorme des vidéos : plusieurs vidéos excèdent une durée d'une heure à un rythme de 24 à 25 images par seconde). Deux approches concurrentes proposent un découpage temporel des documents vidéos : l'approche par *stratification* encore appelée *segmentation au besoin* et l'approche par *structuration hiérarchique* appelée *segmentation a priori*.

La segmentation *a priori* (l'approche que nous adoptons), suppose qu'il existe des unités du document vidéo qui peuvent être mises en place dans un premier temps, l'annotation et la description de ces unités se faisant dans un deuxième temps. L'unité de base considérée est en général le plan, car celui-ci correspond à une unité de montage [6].

L'approche *par stratification* prend pour principe que toute mise en place d'une annotation correspond en même temps à la définition du segment vidéo annoté. Par exemple, c'est le cas lorsqu'un personnage ou une voiture est repéré dans un document, on va définir une strate pour indiquer cette présence [91].

La segmentation *a priori* est basée sur les plans qui peuvent être regroupés en scènes, unités de plus haut niveau sémantique, les scènes peuvent aussi être regroupées en d'autres unités de granularité de plus haut niveau sémantique comme les séquences par exemple. Le résultat de la segmentation est donc un arbre dont le nombre de niveaux de granularité défini dépend non seulement des indexeurs, mais aussi de la taille de la vidéo et de l'application visée. Les différents segments issus de la phase de structuration sont appelés des segments temporels.

Nous allons voir brièvement les concepts sur lesquels se basent les systèmes existants pour partitionner les documents vidéo.

2.3.1 Partitionnement en plans

La segmentation en plan consiste à retrouver la structure temporelle du document vidéo résultant de la phase de montage, en étudiant la variation du signal continu associé aux flux d'images. Le changement brusque de ce signal indique un changement de plan brutal (*cut*), et le changement progressif un fondu (enchaîné, au blanc, au noir) ou un volet. Les types de transitions à considérer dépendent du type de documents traités.

Plusieurs méthodes de détection de plans existent. Certaines méthodes s'appuient sur la comparaison d'histogrammes de couleurs [4, 55, 22] ou sur la comparaison pixel à pixel [51] des images successives, d'autres sont basées sur l'estimation du mouvement [14, 49].

Chacune de ces méthodes possèdent des avantages et des inconvénients. Les méthodes basées sur la comparaison d'histogrammes de couleurs ne sont pas capables de faire la différence entre deux images possédant les même histogrammes mais des contenus différents.

Les méthodes basées sur la comparaison pixel à pixel sont sensibles aux bruits et aux mouvements importants contenus dans les images. De plus, ces méthodes ne tiennent pas compte de l'information spatiale contenue dans les images mais elles sont rapides en temps de calcul. Quant aux méthodes basées sur l'estimation du mouvement, elles sont très lourdes en temps de calcul.

Les différentes méthodes peuvent traiter des séquences vidéo non compressées [50, 51] ou compressées [14, 92, 51] et leurs performances peuvent être évaluées par un taux de fausse détection et un taux d'omission. Le taux de fausse détection mesure le pourcentage de changements de plans faussement annoncé par la méthode utilisée, alors que le taux d'omission mesure le pourcentage du nombre de plans non détectés.

Pour un état de l'art plus complet, le lecteur pourra se reporter à d'autres auteurs [52], notamment pour une comparaison des différentes méthodes basées sur les histogrammes [22] et une comparaison d'algorithmes de détection automatique de transitions brusques et progressives [55]. Ils concluent presque tous que l'utilisation des histogrammes, même simples, donnent toujours de bons résultats. Toutes les méthodes perdent en fiabilité quand il y a mouvement de caméras ou d'objets ou un changement brusque de luminosité.

2.3.2 Regroupement des plans : macro-segmentation

Différentes approches de macro-segmentation ont été proposée. Il n'est pas aisé de donner une vue synthétique de ces approches compte tenu de la diversité des méthodes. Veneau [122] identifie trois grandes familles de techniques :

- les méthodes proposant un regroupement des plans fondé sur une similarité à la fois physique et temporelle de ceux-ci [123, 6] ;
- les méthodes fondées sur l'utilisation d'informations *a priori* [132] ;
- les méthodes fondées sur une utilisation conjointe de différents types d'informations présents dans le document audiovisuel [55].

Le choix de l'une de ces méthodes de macro-segmentation dépend généralement du document vidéo et de l'application visée.

2.3.3 Sélection d'images représentatives

Un problème qui accompagne souvent celui de la segmentation en plans pour l'aide à l'indexation de la vidéo est l'extraction d'images clés, c'est-à-dire d'images les plus représentatives d'un plan. Les méthodes les plus simples consistent à choisir une image prédéterminée : l'image du début, du milieu ou de la fin du plan [95] ou une image suivant

un intervalle de temps fixe [8]. Ces techniques montrent des limites dues aux changements très lents mais significatifs qui peuvent survenir dans les plans. Les techniques utilisés actuellement reposent généralement sur des statistiques liées aux descripteurs utilisés pour le découpage en plan [17].

2.4 Conclusion

Nous avons abordé dans ce chapitre la problématique générale de l'accès à l'information dans des bases de données visuelles. Nous avons en particulier mis l'accent sur les besoins de modélisation puis d'indexation automatique de ces médias. Nous avons en outre dressé un panorama de la variété des modèles existants dans la littérature, en introduisant ponctuellement des besoins d'interaction homme-machine, puisque l'utilisateur est fortement intégré dans le système, de traitement et d'analyse de l'image et de la vidéo pour fournir des capacités d'inférence au système.

Nous avons vu aussi que l'image et la vidéo sont des objets complexes, composé chacun d'un ensemble de caractéristiques pouvant être elles-mêmes complexes.

Dans le prochain chapitre, un état de l'art sur les méthodes de recherche d'information visuelle est présenté. Nous mettons en particulier à profit les méthodes de recherche par la navigation car c'est la navigation que nous utilisons comme moyen de recherche dans des bases de données visuelles. Pour des caractéristiques standard, des critères de similarité seront aussi présentés.

CHAPITRE 3

RECHERCHE D'INFORMATIONS VISUELLES PAR LE CONTENU

La recherche dans des bases de données normalisées (utilisant des SGBD relationnels, à objets, etc., et n'utilisant que des attributs simples) est bien maîtrisée. Par contre, la recherche dans les bases de données *multimédias* pose plusieurs problèmes : les principaux sont la définition de données de haut niveau et la définition de requêtes. En effet, la définition de la requête est tout aussi importante que celle des données. Plus précisément, le choix d'un couple de représentation pour les requêtes et les données conditionne le développement d'un système de recherche d'information.

Les problèmes liés à la définition de données de haut niveau ont été présentés au chapitre 2. Dans ce chapitre, nous présentons les problèmes liés à la définition de la requête en présentant plusieurs formes d'interrogation que l'on trouve dans différents prototypes. Nous mettons davantage l'accent sur la navigation.

3.1 De l'interrogation par requête à la recherche par navigation

Les données multimédias (texte, image, vidéo et son) contiennent une densité variable d'informations et beaucoup de redondances. Un utilisateur cherchant un média s'attend à trouver des informations sémantiques alors qu'on ne peut, dans la plupart des cas, n'extraire directement que des données structurelles. En plus, l'utilisateur ne sait pas toujours exprimer sa requête dans un langage formel. On peut par exemple chercher des photographies en gros plan de fleurs, dans ce cas il s'agit d'images sur fond vert ayant la texture connue de l'herbe avec un objet circulaire de couleur plus ou moins vive. Il peut aussi chercher des photographies de chevaux, dans ce cas un cheval alesan (de robe brune) dans une prairie

sera aussi pertinent qu'un cheval blanc dans une carrière. Cette fois les couleurs ne sont plus discriminantes, seule la forme permet de trouver les images souhaitées. Ainsi deux requêtes similaires sémantiquement seront très différentes formellement.

Il est donc important de définir une interface intuitive permettant à l'utilisateur d'atteindre l'ensemble des objets qu'il recherche.

Dans la suite, nous présentons différentes formes de recherche que l'on rencontre dans la littérature. La recherche par requêtes formelles voire par rétroaction a fait l'objet de nombreux travaux [117, 26, 32]. En revanche, la recherche par navigation semble avoir été beaucoup moins étudiée. Or, la navigation, en s'appuyant sur une organisation pertinente des données, rend la recherche dans une base plus aisée et efficace qu'en effectuant des requêtes [78]. Par conséquent dans ce chapitre, nous mettons l'accent sur les méthodes de navigation rencontrées dans la littérature.

3.1.1 Recherche par requêtes formelles

L'interrogation *formelle* met l'accent sur la *spécification* de la classe de documents à retrouver, c'est-à-dire fondamentalement sur la construction d'une requête (même si le langage d'interrogation peut être très frustre, se ramenant même à une simple fonction linéaire)

L'interrogation formelle peut se faire aussi bien *via* un langage de requêtes (textuel) que sous forme graphique, donc interactive. En tout état de cause, le principe se résume à spécifier la description d'un document virtuel, c'est-à-dire la gamme de valeurs qu'un ou plusieurs attributs doivent satisfaire, l'ensemble étant généralement lié par une conjonction implicite.

3.1.1.1 Requêtes textuelles

Les requêtes textuelles offrent à un utilisateur la possibilité de formuler sa requête à l'aide de langages de requêtes tels que SQL [117]. La recherche se fait sur un ensemble de mots clés associés manuellement au contenu des documents, ce qui ramène les bases de données multimédias à des bases de données textuelles. Les difficultés rencontrées lors de ce processus sont nombreuses. Le principal problème découle du fait que les descripteurs fournis par les techniques d'analyses d'images ne fournissent pas un niveau d'abstraction sémantique suffisant pour tout document ce qui conduit à des niveaux de description et d'interprétation des documents visuels variables d'un descripteur à un autre et d'un document à un autre. Aussi, l'indexation manuelle d'une base contenant plusieurs images est longue et fastidieuse. Par conséquent, les systèmes basés sur des requêtes textuelles demeurent li-

mités et sont généralement associés à des familles de documents visuels spécifiques comme les journaux télévisés. La voie la plus prometteuse dans ce contexte concerne les vidéos et consiste à transcrire la bande son pour en déterminer le sujet [53, 127] plutôt qu'à exploiter le contenu visuel.

3.1.1.2 Requêtes graphiques

La forme d'*interrogation graphique* est une forme plus agréable et généralement simplifiée de rédaction de la requête [15, 60, 104]. Elle peut être quasiment incontournable dans certain cas, comme pour spécifier précisément une texture en la choisissant dans une palette plutôt qu'en décrivant ses caractéristiques numériques [26].

Malgré la simplicité relative de cette approche « classique », deux nouveautés apparaissent déjà. Tout d'abord, il est préférable de passer d'une interrogation stricte à une *interrogation approximative*. Et, il se pose des problèmes aigus d'*indexation multidimensionnelle*.

On distingue deux formes principales d'interrogation graphique : les requêtes par croquis et par l'exemple.

Requêtes par croquis : Les requêtes par croquis et spécification de caractéristiques consiste à donner à l'utilisateur la possibilité de formuler sa requête par le biais de dessins tout en spécifiant les caractéristiques visuelles et les relations spatio-temporelles liées aux objets contenus dans les documents (par exemple « un disque rouge sur un fond vert » pour retrouver des images de roses [26]). Initialement considérée dans le cas des images fixes, cette technique de recherche a été étendue aux vidéos par ajout d'informations de nature dynamique. Un croquis est défini comme une image où l'utilisateur peut assigner des mouvements à n'importe quelle partie d'une scène. Ce mouvement est réalisé à l'aide d'objets vidéo définis comme une collection de régions visuelles groupées sur quelques critères à travers plusieurs trames. Les régions sont affectées de plusieurs attributs tel que la couleur, la texture, la forme, le mouvement et le temps. Dans [102] le mouvement et la durée temporelle sont les attributs principaux assignés à chaque objet dans le croquis en plus des attributs habituels tels que la forme, la couleur et la texture. En utilisant une palette visuelle, les utilisateurs esquissent une scène en dessinant une collection d'objets visuels. Les caractéristiques spatio-temporelles de ces objets définissent la scène dans son ensemble.

Requêtes par l'exemple : Les systèmes de recherche par l'exemple sont ceux qui permettent de traiter la plus grande variété de contenus visuels [44, 56, 111]. Cette approche

consiste à soumettre un exemple d'une image d'un plan ou d'une autre unité de granularité faible à la base de données visuelles et la phase de recherche consiste à retrouver des unités de contenus similaires à la requête. La phase d'indexation pour ce type de requêtes consiste à extraire un certain nombre de paramètres d'analyse d'images (couleur, texture, forme...), qui sont à même de caractériser de manière discriminante les unités de granularité présentes dans le média. Ces unités sont alors représentées dans un espace de référence par la valeur de ses paramètres.

Lors du processus de recherche, l'unité de requête sera placée dans le même espace de référence que les unités indexées. Ainsi les unités les plus proches de l'unité de requête dans cet espace de référence (ayant des valeurs paramétriques similaires) sont présentées en résultat à l'utilisateur comme les plus proches visuellement de l'unité de requête.

Contrairement aux méthodes textuelles et par croquis, les méthodes de recherche par l'exemple permettent de répondre à des questions ne faisant pas appel à un cadre sémantique. Les difficultés rencontrées avec ces méthodes sont principalement liées au fait qu'il n'est pas aisé de rendre compte de la perception visuelle par des paramètres d'analyse d'images. Il peut ainsi exister un certain déphasage entre ce que la méthode de recherche considère comme des unités similaires d'un point de vue paramétrique et ce que l'œil de l'utilisateur considère comme des unités similaires d'un point de vue visuel.

3.1.2 Recherche par rétroaction

La requête initiale d'un utilisateur ne décrit souvent pas très bien son besoin d'information. Les requêtes sont dénuées de concepts importants. L'idée de la rétroaction est de capter avec précision le besoin de l'utilisateur à travers des documents qu'il a jugé pertinents. Le principe est le suivant : après avoir lancé une requête et reçu les réponses, l'utilisateur, en visualisant les réponses peut indiquer aux systèmes celles qui sont pertinentes à son sens, et celles qui ne le sont pas. Avec ces indications, le système peut reformuler la requête. Progressivement, la reformulation de la requête sépare de mieux en mieux les documents pertinents pour l'utilisateur de ceux qui ne le sont pas.

La rétroaction peut être appliquée aux différents types de requêtes cités ci-dessus. Le principal problème réside dans le fait que la requête est évaluée à chaque fois qu'elle est reformulée.

Les différentes formes de recherche d'information visuelle que nous venons de voir ont tous des avantages et des inconvenients que nous avons presenté. La méthode de recherche par navigation que nous présenterons dans la section 3.1.3 est une nouvelle approche différente. La base d'images ou de vidéos n'est plus un simple ensemble dont les éléments

sont ordonnées suivant la requête effectuée, mais un espace (de dimension quelconque) où sont classées les images ou les vidéos suivant leurs similitudes. La recherche est donc un *voyage* à travers cet espace. L'utilisateur n'a plus besoin de préciser ce qu'il cherche : il n'a qu'à naviguer en se dirigeant toujours vers les images ou les vidéos qui lui plaisent le plus. Il pourra ainsi se déplacer suivant des critères différents (forme, couleur, texture...) sans même connaître ces notions.

3.1.3 Recherche par navigation

La navigation est basée sur l'exploration d'une structure d'arbre ou de graphe. Cette forme d'interaction tire profit d'une caractéristique importante de la cognition humaine : il est plus facile de reconnaître quelque chose d'intéressant que de le décrire.

En effet, la navigation est une technique issue des travaux sur les hypertextes qui a démontré, lorsqu'elle s'appuie sur une organisation pertinente des données, que la recherche dans une base était plus aisée et même plus efficace qu'en effectuant des requêtes [78]. Elle consiste en une classification (automatique ou semi-automatique) hors-ligne des documents selon des critères bien précis. La phase de navigation consiste donc à offrir à l'utilisateur la possibilité de naviguer dans une classification.

Nous verrons dans cette section quelques techniques de navigation dans des bases de données vidéos en essayant de faire ressortir leur apport principal.

Les techniques de recherche de documents vidéo présentées plus haut sont largement utilisées dans la littérature [32, 117, 44, 56]. La navigation est une méthode qui semble pourtant plus naturelle. Elle donne une idée du contenu d'une base de données, et permet de retrouver rapidement les entités désirées sans résolution effective d'une requête.

Les méthodes de navigation rencontrées sont nombreuses. Comme pour la modélisation, nous avons dressé le tableau 3.1 qui offre une vue synthétique et résumée des propositions référencées, sous plusieurs points de vue. Il classifie les propositions en fonction des éléments de navigation et des types de navigation utilisés. Aussi, pour faciliter la comparaison, les modèles ont été traduits dans le formalisme UML.

La navigation intra vidéo consiste à naviguer sur une vidéo. Elle est basée sur une classification hiérarchique rendant compte des similarités entre signatures et permet de visualiser une vidéo sous la forme d'un arbre ou d'un graphe de voisinage. Elle est définie grâce à différents éléments de navigation qui sont généralement transposés pour la définition de liens pour la navigation inter vidéos. La différence étant qu'au lieu de rechercher les similitudes dans une même vidéo, la recherche est effectuée sur l'ensemble des vidéos de la base.

Une manière simple de réaliser cette forme de navigation est de structurer la vidéo en

éléments de navigation	navigation intra vidéo	navigation inter vidéos
plans	**[94]**	[80]
images clés	[8], **[94, 93]**	[34], **[94, 93]**
objets	[13], **[46]**	[35], **[46]**
mots clés	[13],**[46, 35, 94]**	[31], **[46, 35]**
storyboard	[20]	**[93]**
arbre de structuration	**[94]**	
genres		[2, 42, 120]

Table 3.1 – Éléments et types de navigation

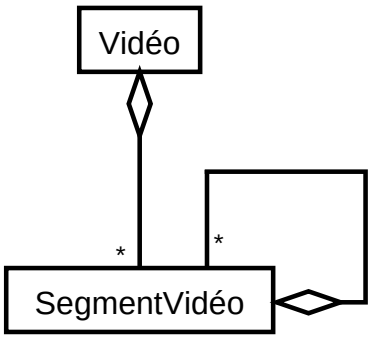

Figure 3.1 – Schéma UML pour la navigation hiérarchique sur les segments vidéo

segments temporels (le plan peut être le niveau le plus bas de cette structuration) et d'organiser ces segments sous forme d'arbre de manière à refléter la structuration hiérarchique de la vidéo, permettant de naviguer sur l'arborescence sans résolution effective d'une requête [94, 80]. Le schéma de la figure 3.1 illustre le formalisme UML de ce modèle.

Dans une telle organisation, la navigation sur la vidéo se fait sans avoir une idée préalable du contenu des segments qui pourrait guider l'utilisateur dans le choix des segments pour la navigation.

Pour pallier à ce problème, certains auteurs proposent l'extraction d'une ou de plusieurs images clés par segment qui peuvent être décrites automatiquement par des descripteurs de bas niveau ou manuellement par des descripteurs textuels (cf. figure 3.2). Ces images sont ensuite affichées sur un panel de manière à offrir un aperçu du contenu des segments. Il est possible de cliquer sur une image afin de voir la vidéo à partir de cette image, ou bien de choisir une image exemple afin d'accéder à des images similaires [8, 93]. Les images peuvent aussi être organisées de manière à offrir une technique de navigation latérale. Ainsi,

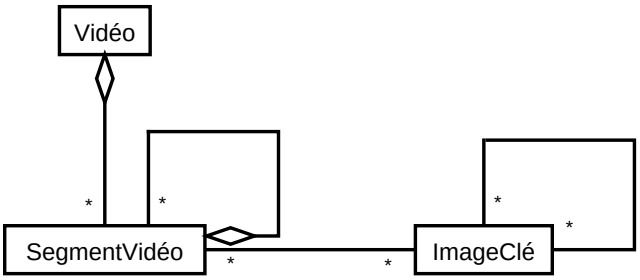

Figure 3.2 – Schéma UML pour la navigation sur les images clés

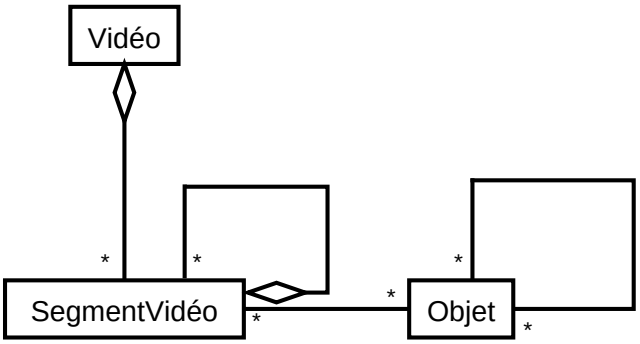

Figure 3.3 – Schéma UML pour la navigation sur les objets

la méthode NN^k de Heesch *et al.* [34] réalise hors ligne la liaison entre images similaires (appelées plus proches voisins) en fonction d'un nombre k de caractéristiques ce qui permet ensuite de présenter les images sous la forme d'un réseau, où un clic sur chaque nœud du réseau affiche à son tour les plus proches voisins de ce dernier.

D'autres auteurs proposent d'utiliser les caractéristiques des objets contenues dans les segments vidéos et les relations existantes entre ces caractéristiques (cf. figure 3.3) pour naviguer sur les vidéos. Ces caractéristiques peuvent être des caractéristiques de bas niveau telles que la couleur ou la forme. La navigation se fait par clic sur des objets apparaissant dans des segments vidéos [46, 35, 13].

Des modèles utilisant des *storyboards* annotés [20] ou des mots clés associés aux objets [13] ont aussi été proposés. Les mots clés permettent de créer des hyperliens entre objets apparaissant dans des segments vidéos [94, 46, 35, 13]. Les mots clés de même type peuvent être regroupés dans une classe et le nom de la classe affiché à côté de l'objet apparaissant dans le segment. Un clic sur un mot clé représentant le nom d'une classe affiche la liste des mots clés de cette classe et permet de choisir un élément de la liste afin de naviguer vers des éléments similaires. Le schéma de la figure 3.4 illustre le formalisme UML de ce modèle.

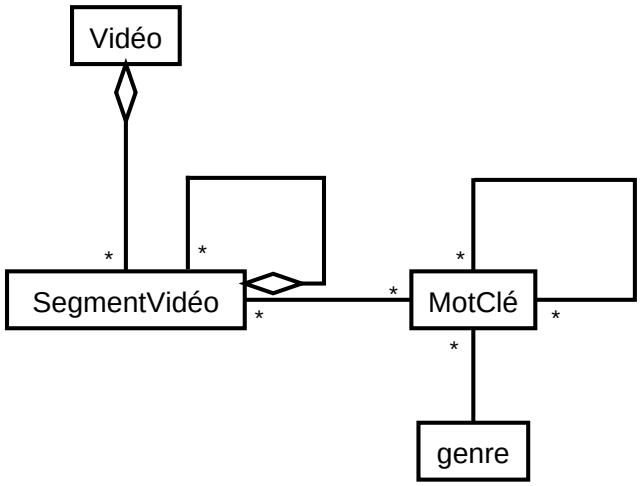

Figure 3.4 – Schéma UML pour la navigation sur les mots clés

Dans le modèle de Christel et Warmack01 [20], les *storyboards* sont créés par la détection de plans à l'aide d'histogrammes de couleurs et la représentation de chaque plan par une image clé déterminée par le mouvement de la caméra et la détection de visages. Pour l'annotation, les paroles sont analysées par Sphinx-III recognizer [124] et les textes sont transcrits en bas de chaque image clé. Le processus de recherche débute par une recherche textuelle sur les annotations associées aux plans, et la navigation sur les images clés représentant les plans retournés par la requête textuelle.

Les hyperliens pour la navigation intra vidéo et inter vidéos peuvent être mis en place manuellement [35] ou bien calculés [13, 54]. Leur organisation, pour offrir une structure de navigation conviviale à l'utilisateur, est une étape importante dans la mise en place d'un système de recherche d'information par navigation.

Outre l'organisation des éléments vidéos, certaines méthodes réalisent la classification automatique des documents vidéos en genres (journaux télévisés, retransmission sportives, types de films...) [2, 42, 25], ce qui permet de proposer de naviguer dans une arborescence de catégories de manière analogue à certains moteurs de recherche. Ainsi des applications se sont attachées à déterminer des descripteurs permettant de différencier les films d'action et de romances [120, 42]. Ces techniques de classification nécessitent un fort degré d'interactivité avec l'indexeur pour être valides, d'où le besoin de proposer des alternatives moins ambitieuses mais automatisables.

3.2 Mesures de similarité

Dans le chapitre 2, les caractéristiques incontournables ont été présentées, la section 3.1 a présenté le problème de la définition de la requête. Nous présentons dans cette section la construction de mesures de similarités entre documents indexés et les requêtes des utilisateurs.

Les premières ressemblances mathématisées sont les transformations géométriques, trop rigides. À l'extrême opposé, les ressemblances topologiques permettent de rendre équivalents une tasse et un anneau. Pour traiter les ressemblances, il convient de bâtir le concept sur des mesures, c'est-à-dire d'introduire de la gradualité.

Une mesure de similarité est souvent définie au travers d'une distance :

$$d : C \times C \rightarrow \mathbb{R} \tag{3.1}$$

vérifiant trois axiomes :

- auto-similarité : $\forall x, d(x,x) = 0$;
- symétrie : $\forall (x,y), d(x,y) = d(y,x)$;
- inégalité triangulaire : $\forall (x,y,z), d(x,y) + d(y,z) \geq d(x,z)$.

Les distances sont nombreuses dans la littérature, définies pour des valeurs scalaires, ensemblistes, vectorielles, etc. (différence absolue, cosinus, Harman, Dice, Jacquard, degré d'inclusion, produit scalaire, Manhattan, Hamming, euclidienne, Hausdorff, informationnelle...).

Si la distance est normalisée, alors une similarité peut être simplement calculée comme son complément à un : $s(c,c') = 1 - d(c,c')$.

Malheureusement, plusieurs expériences ont montré que la perception humaine n'est pas assimilable à une distance. Souvent, elle ne vérifie ni la symétrie, ni même l'auto-similarité [99].

De manière plus générale donc, chaque fois que l'on pourra plonger une propriété perceptuelle dans un espace métrique, il faudra en déduire une similarité comme une fonction non triviale, monotone non décroissante, f, sur une distance :

$$s : \begin{array}{ll} C \times C \rightarrow & [0,1] \\ (c,c') \mapsto & f(d(c,c')) \end{array} \tag{3.2}$$

Pour les documents vidéos, il convient de tenir compte de leur aspect temporel. La définition de mesure de similarité repose souvent sur une caractérisation globale des plans.

Liu *et al.* [56] définissent quatre critères de la mesure humaine de la similitude vidéo qui sont :

- La similarité visuelle : deux vidéos similaires doivent avoir des caractéristiques visuelles de bas niveau similaires. Généralement c'est le principal critère de la mesure de similarité entre vidéos.
- La similarité de l'ordre temporel : Deux vidéos peuvent avoir des plans semblables mais dont les contenus sont ordonnés différemment. Généralement, on considère que ces deux vidéos ne sont pas similaires du point de vue de la mesure humaine de la similarité.
- La similarité de la durée temporelle : Un même film peut être éditer avec différentes durées. Ce critère ne doit pas influencer sur la mesure de similarité entre les différentes versions de cette vidéo.
- La similitude de granularité : C'est la similarité entre les différentes parties issues de la segmentation des vidéos. Idéalement, si deux vidéos sont similaires, elles doivent avoir le même nombre de plans, similaires un-a-un. Mais cela est rarement le cas car, comme nous l'avons souligné, des vidéos semblables peuvent avoir des durées temporelles différentes dues aux effets de montage.

Généralement, seuls le premier critère est pris en compte en raison des grandes diversités de montage et de correspondances des structures vidéos car si les plans vidéos semblent tendre à préserver l'ordre temporel des trames semblables, cela est trop restrictif pour la scène vidéo due au fait que de nombreuses scènes vidéos semblables ont des plans ordonnés différemment à cause des effets de montage. Ce critère est suffisant et permet de capturer l'information temporelle et spatiale de la vidéo [56]. Par exemple étant donné une vidéo composée d'une scène comportant trois plans P_1, P_2, P_3 disposés dans l'ordre temporel suivant : $P_1P_2P_3$, cette même scène peut être présente dans une autre vidéo mais avec des effets de montage comme P_3P_2.

Ces mesures permettent de mieux capturer l'information temporelle et spatiale de la vidéo [57].

3.3 Architecture

À partir de ces généralités, on se rend compte que les points à éclaircir sont : la représentation des médias, les mesures de similarités, les méthodes d'indexation physique, la prise en compte de l'utilisateur dans une boucle de rétroaction, et, plus récemment, la classification.

Le cœur d'un système de recherche d'information visuelle doit donc être flexible [130]. D'une part, il doit s'adapter à des classes d'applications particulières utilisant des informa-

tions nouvelles, différentes, connues *a priori* ou encore apprises, ce qui nécessite d'envisager sa construction dans une optique très ouverte. D'autre part, il lui faut découvrir les objectifs des utilisateurs afin d'améliorer la pertinence des réponses, y compris, en définitive, pour des utilisateurs avertis.

D'un point de vue technique, l'architecture retenue ne doit pas être un facteur de handicap. En effet, le module doit pouvoir se coupler aussi bien au sein d'applications de bases de données que sur des systèmes plus ouverts, par exemple des applications de vision artificielle.

3.4 Conclusion

Dans ce chapitre, nous avons distingué trois modes de recherche complémentaires qui sont (1) la recherche formelle par requête, adaptée au développement d'applications, (2) la recherche par inter-action entre l'utilisateur et le système, indispensable pour les interrogations *ad hoc*, et (3) la navigation au sein d'une base de données préalablement organisée. Si pour les SGBD traditionnels et les systèmes de recherche d'information textuelle, la sémantique d'une requête basée sur le contenu reste relativement simple, car elle consiste à trouver les données correspondant aux mots clés spécifiés dans la requête. Nous avons vu que pour les données visuelles, l'expression d'une requête et l'accès aux données deviennent bien plus difficiles. Cependant les trois formes ne sont pas exclusives puisque (2) s'appuie sur (1), que (2) et (3) intègrent l'utilisateur dans le processus de recherche et peuvent être utilisées en alternance dans une même séance, et qu'enfin (3) peut être utilisée dans (1) lorsque la classification a été stockée dans la base.

PARTIE II

Proposition

La première partie nous a permis d'exposer les problématiques liés à l'indexation et à la recherche des documents visuels. Nous avons étudié les principaux modèles disponibles et avons souligné les limites.

Cette deuxième partie nous permet de décrire nos propositions et de montrer en quoi elles répondent mieux aux besoins exprimés.

Le chapitre 4 rappelle les résultats obtenus pour $Click^{Im}_{AGE}$ pour la modélisation des images.

Le chapitre 5 détaille le modèle $Find^{Vi}_{DEO}$ et précise comment il répond aux besoins d'expressivité et de flexibilité.

Le chapitre 6 présente notre proposition sur la navigation dans une base d'images et de vidéos. Nous y présentons une technique de navigation qui permet de naviguer en basculant indistinctement entre images et vidéos.

Dans le chapitre 7, nous présentons l'implémentation relationnelle de notre modèle. Les tests effectués montrent que cette implémentation répond au besoin de performances.

CHAPITRE 4

MODÉLISATION DE L'IMAGE

Dans ce chapitre, nous présentons le système $Click^{Im}_{AGE}$ developpé par notre équipe de l'université de Nantes [64]. Pour $Click^{Im}_{AGE}$, la representation des données est semi-structurée, et la description retenue est un ensemble de métriques basées sur le contenu de l'image et classées suivant le modèle MPEG-7. En effet, MPEG-7 définit une norme de description des contenus audiovisuels afin d'en simplifier l'indexation et la recherche. Les outils de description fournis par MPEG-7 sont en mesure de préciser un grand nombre d'informations supplémentaires, pouvant être classées en six catégories principales :

- (i) - les informations de format (images en couleur, en niveaux de gris, en infra-rouges...) ;
- (ii) - les informations physiques (couleurs principales, textures...) ;
- (iii) - les informations perceptuelles (couleurs chaudes, texture grossière...) ;
- (iv) - les informations structurelles (régions d'une image...) ;
- (v) - les metadonnées intrinsèques (« objet » du monde réel associé à une région, mots clés...) ;
- (vi) - diverses annotations.

4.1 Modélisation des images

Dans $Click^{Im}_{AGE}$, les niveaux (i) à (iv) de MPEG-7 sont utilisés. La représentation des données est un ensemble de métriques basées sur le contenu des images. Ces métriques sont principalement :

- des informations sur la forme générale de l'image (taille, orientation, élongation) ;
- des informations de couleur classées par zones : les couleurs sont issues d'une segmentation de l'espace à partir du repère HSV, ces informations de couleur sont associées aux images en utilisant la logique floue.

À partir de ces métriques, à chaque image est associé un ensemble d'attributs formant une relation binaire entre les images et ces attributs. Puis, à partir de cette relation binaire, est calculé un treillis de Galois, structure utilisée pour la navigation.

Définition 4.1 (Espace de description)**.** L'espace de description est une union de plusieurs sous-espaces de description défini par :

$$\mathcal{D} = \mathcal{D}_{\text{surface}} \cup \mathcal{D}_{\text{orientation}} \cup \mathcal{D}_{\text{élongation}} \cup (\text{region} \times \mathcal{D}_{\text{teinte}}) \cup (\text{region} \times \mathcal{D}_{\text{saturation}}) \cup (\text{region} \times \mathcal{D}_{\text{intensité}}) \quad (4.1)$$

4.1.1 Étiquettes linguistiques floues pour la couleur

Les étiquettes linguistiques floues (ELF) telles que « rose » sont décrites sur l'espace de couleur HSV *(Hue, Saturation, Value)*. Chaque ELF c est associée à une fonction d'appartenance μ_c dont les valeurs sont comprises dans [0,1], reflétant la manière dont l'ELF c décrit la couleur d'une image ou d'une région d'une image. Par exemple, l'étiquette « rose » peut être représentée dans l'espace de couleur HSV par une fonction d'appartenance $\mu_{rose}(h, s, v)$ avec une grande valeur d'appartenance associée aux couleurs avec une teinte limitée par le rouge, une saturation faible et une valeur d'intensité assez élevée.

Définition 4.2 (sous-ensemble flou)**.** Un sous-ensemble flou A d'un ensemble X est définie par une fonction d'appartenance (ou fonction caractéristique) μ_A où $\forall x \in X, \mu_A(x) \in [0, 1]$. $\mu_A(x)$ représente le dégré d'appartenance de x au sous-ensemble flou A

Cette approche générale [97] permet de représenter chaque ELF par un ensemble flou défini sur le domaine de définition de la couleur. La couverture de l'espace de couleur est requise pour assurer que chaque image aura au minimum une représentation avec des descripteurs linguistiques. Permettre qu'une ELF puisse être définie indépendamment des autres ELF permet une grande flexibilité, mais le domaine de couverture doit être vérifié après que toutes les définitions d'étiquettes aient été placées. Avec cette méthode, il est possible que des ensembles restent incomplets.

Pour surmonter ce problème, une autre méthode est utilisée. Chaque étiquette linguistique de couleur c est formée par la concaténation d'étiquettes élementaires $c.h$, $c.s$ et $c.v$ définie pour chaque H, S et V. La fonction d'appartenance associée à c est calculée comme une conjonction des fonctions d'appartenance d'étiquettes élémentaires :

$$\mu_c(h, s, v) = \mu_{c.h}(h) \otimes \mu_{c.s}(S) \otimes \mu_{c.v}(v) \quad (4.2)$$

où $\otimes$ représente une T-norme ($\max$ par exemple) utilisée pour calculer la conjonction des canaux individuels.

À partir de cette représentation, le partitionement suivant a été choisi :

- teinte : rouge, orange, jaune, vert, cyan, bleu et magenta ;
- saturation : vive et terne ;
- valeur (intensité) : sombre et claire.

Les étiquettes sont ainsi formées par la combinaison de ces termes (comme : rouge vif clair). À ces couleurs viennent s'ajouter quatres couleurs particulières qui sont traitées à part à cause du défaut classique de HSV (la teinte n'est définie que si la saturation est au dessus d'un certain seuil et celle-ci dépend de la luminance). En dessous d'un certain niveau d'intensité, la couleur est perçue comme noire. Aussi, sous un certain niveau de saturation la couleur est perçue comme grise ou blanche. Compte tenu de ces constatations, le noir, le gris sombre, le gris clair, et le blanc on été ajoutés, ce qui donne au total 32 termes.

Cette méthode est simple du point de vue de l'utilisateur et la couveture de l'espace de couleur HSV est garantie tant que chaque dimension est couverte, ce qui rend l'implémentation facile.

4.1.1.1 Caractérisation de la zone de couleur

La perception de la couleur résulte de la juxtaposition des pixels. Considérant la représentation linguistique de couleurs retenue, chaque pixel de couleur est exprimé en terme d'étiquette de couleur avec différents poids. Pour un pixel, le poids est le degré d'appartenance de sa couleur à l'ensemble flou associé à l'étiquette couleur. Par exemple, le pixel « rose » peut être défini comme formé de deux étiquettes de couleur :

- rouge vif clair avec un dégré membre de 0,1 ;
- rouge terne clair avec un dégré membre de 0,9.

Étant donné une région S défini comme un groupe de pixels adjacent, l'importance relative $\tau(d)$ d'une étiquette de couleur d dans S, est calculée comme la somme des degrés d'appartenance $\mu_d(p)$ de chaque pixel :

$$\tau_S(d) = \frac{\sum_{p \in S} \mu_d(p)}{\sum_{d' \in D} \sum_{p \in S} \mu_{d'}(p)}$$

où D est l'ensemble des étiquettes de couleur.

Figure 4.1 – Découpage en cinq parties trapézoïdales

4.1.2 Division syntaxique

Une image contient plusieurs objets du monde réel, la séparation sémantique de ces objets offre une description plus précise des images couleur, cela améliore considerablement la qualité des résultats. Des algorithmes de segmentation effectuent cette séparation par identification de zones homogènes utilisant des informations de couleur et souvent des informations de texture.

Cependant, dû au coût élevé de ces algorithmes et à leur faiblesse pour reconnaître des objets du monde réel, $Click^{Im}_{AGE}$ adopte une division syntaxique des images plutôt qu'une réelle segmentation.

Généralement les images photographiées présentent l'objet principal au centre de l'image et le pourtour représente le décor. Ces parties représentent donc des informations bien différentes. En plus, l'homogénéité des couleurs augmente si les zones choisies sont petites. Par exemple dans une image de paysage, le ciel est suceptible d'être bleu avec quelque zones grises et le sol probablement vert. Pour capter ces informations, la segmentation en cinq parties trapézoïdales a été utilisée. La figure 4.1 montre un exemple de « segmentation ».

Les cinq trapèzes sont respectivement les parties centrale, gauche, droite, haute et basse de l'image. On remarque l'objet principal au centre. Le centre couvre plus de 40 % de la surface totale de l'image. Les autres trapèzes représentent seulement 15 % chacun.

4.1.3 Mesures géométriques générales

En plus de la couleur, des mesures géométriques sont choisies pour représenter la surface, l'orientation, et l'élongation d'une image.

L'utilisation de l'orientation et de l'élongation pour la recherche d'image est tout à fait évidente : une image plus large que haute est généralement qualifiée d'image « paysage » parce que la plupart des images de paysage ont cette proportion. Souvent, même si ce n'est pas un paysage réel, une image orientée en paysage représente usuellement plusieurs objets du monde réel. En revanche, une image plus longue que large est appelée une image « portrait » parce que les portraits ont généralement cette proportion. Ainsi, il existe une certaine corrélation implicite entre l'orientation, l'élongation et la sémantique de l'image.

La surface est aussi une mesure importante car des utilisateurs peuvent rechercher des images avec une certaine taille en fonction de leur besoin.

4.1.3.1 Caractéristiques numériques

Premièrement, D_{surface}, $D_{\text{orientation}}$ et $D_{\text{élongation}}$ sont des informations sur le format (MPEG-7 niveau i), liées à une image i et basées sur les fonctions scalaires suivantes :

- $\text{surface}(i) = \text{largeur}(i) \times \text{hauteur}(i)$;
- $\text{orientation}(i) = \frac{1-\cos(2\alpha(i))}{2}$;
- $\text{élongation}(i) = |\frac{4\alpha(i)}{\pi} - 1|$ où $\alpha(i) = \arctan \frac{\text{largeur}(i)}{\text{hauteur}(i)}$ est l'angle de la diagonale de l'image.

L'élongation est représentée directement par l'angle tandis que l'orientation est une fonction de l'élongation. L'utilisation de la valeur absolue permet de décorréler ces deux valeurs : la covariance est nulle sur $[0, \frac{\pi}{2}]$. Intuitivement, cela semble encore insuffisant. En effet, la corrélation est parfaite sur les intervalles $[0, \frac{\pi}{4}]$ et $[\frac{\pi}{4}, \frac{\pi}{2}]$ pris indépendamment. Ce qui découle du fait que l'orientation dépend fonctionnellement de l'élongation

Deuxièmement, D_{teinte}, $D_{\text{saturation}}$ et $D_{\text{intensité}}$ sont liés en même temps aux informations physiques et perceptuelles parce que c'est un espace de couleurs perceptuel qui est utilisé, car corrélé au langage naturel (MPEG-7 niveau iii).

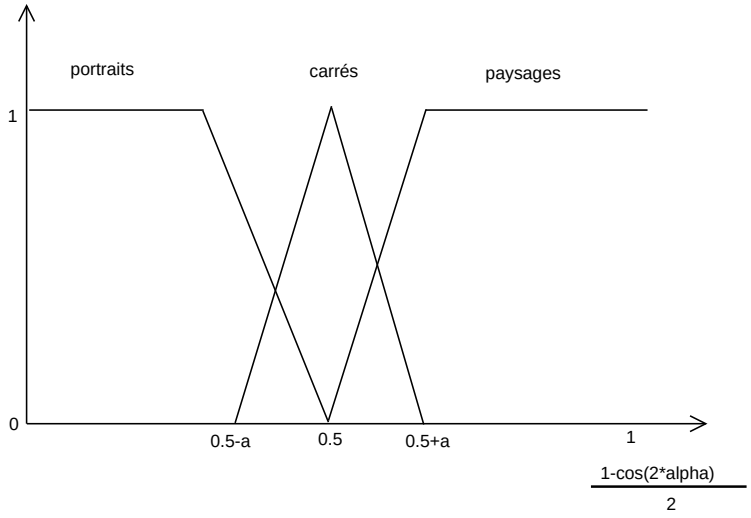

Figure 4.2 – Le descripteur « orientation » et les trois sous-ensembles flous qui caractérisent la variable linguistique correspondante

4.1.3.2 Variables linguistiques

Les caractéristiques numériques citées sont utilisées pour associer des métadonnées aux images sous formes de sous-ensemble de descriptions :

$$\mathcal{D}_{\text{surface}} = \{minuscule, petit, moyen, large, énorme\} \tag{4.3}$$

$$\mathcal{D}_{\text{orientation}} = \{portrait, carré, paysage\} \tag{4.4}$$

$$\mathcal{D}_{\text{élongation}} = \{rien, standard, panoramique, allongé\} \tag{4.5}$$

$$\mathcal{D}_{\text{teinte}} = \{rouge, orange, \ldots, cyan, bleu, magenta\} \tag{4.6}$$

$$\mathcal{D}_{\text{saturation}} = \{vive, claire, pâle\} \tag{4.7}$$

$$\mathcal{D}_{\text{intensité}} = \{noire, sombre, claire, blanche\} \tag{4.8}$$

Ces sous-ensembles discrets sont obtenus à partir des variables linguistiques floues et des sous-ensembles flous correspondants, ainsi que par des seuillages appropriés.

La figure 4.2 illustre une manière simple de définir des sous-ensembles flous d'une variable linguistique, c'est-à-dire que pour chaque sous-description $\mathcal{D}_i$ on associe une fonction d'appartenance arbitraire.

Définition 4.3 (Valeur floue d'un scalaire)**.** La valeur floue d'une propriété scalaire dependant d'un sous-ensemble flou A est directement donnée par sa fonction d'appartenance :

$$FV_i : \begin{array}{l} \mathbb{R} \times \mathcal{D}_i \rightarrow [0,1] \\ (x, A) \mapsto \mu_A(x) \end{array} \tag{4.9}$$

où $i \in \{\text{surface}, \text{orientation}, \text{élongation}\}$.

Définition 4.4 (Valeur floue d'un vecteur)**.** La valeur floue d'un histogramme h dependant d'un sous-ensemble A est calculée comme suit :

$$FV_j : \begin{array}{l} ([0,1] \rightarrow [0,1]) \times \mathcal{D}_j \rightarrow [0,1] \\ (h, A) \mapsto \int_0^1 h(x) \times \mu_A(x)\, dx \end{array} \tag{4.10}$$

Définition 4.5 (Description floue)**.** Une description floue est un ensemble de noms de sous-ensembles correspondant à des valeurs floues non-nulles.

$$FD_i : \begin{array}{l} \mathbb{R} \cup ([0,1] \rightarrow \mathbb{N}) \rightarrow 2^{\mathcal{D}_i \times [0,1]} \\ p \mapsto \{(A, FV_i(p, A)) | A \in \mathcal{D}_i \wedge FV_i(p, A) > 0\} \end{array} \tag{4.11}$$

4.1.3.3 Modèle binaire

Pour pouvoir utiliser le treillis de Galois, les degrés d'appartenance flous sont supprimés et une relation binaire entre images et métadonnées est produite.

Définition 4.6 (Description discrète)**.** La description discrète est obtenue à partir d'une description floue et de seuils α_i :

$$DD_i : \begin{array}{l} \mathbb{R} \cup ([0,1] \rightarrow \mathbb{N}) \rightarrow 2^{\mathcal{D}_i} \\ p \mapsto \{A | (A, f) \in FD_i(p) \wedge f \geq \alpha_i\} \end{array} \tag{4.12}$$

Par exemple, l'élongation d'une image pourrait être (standard, 0,3), (panoramique, 0,7) et mènerait à {standard, panoramique } si la coupe α est placée à $0,3$. De même, à partir de {(rouge, 0,5), (orange, 0,1), (bleu, 0,4), (magenta, 0,2)}, on obtient {rouge, bleu, magenta} avec un seuil fixé à $0,2$.

Définition 4.7 (Métadonnées discrètes)**.** Finalement, la description complète d'une image est donnée par :

$$M : \begin{array}{l} \mathcal{I} \rightarrow \in^{\mathcal{D}} \\ i \mapsto \bigcup\limits_{d \in general} DD_d(d(i)) \cup \\ \qquad \bigcup\limits_{r \in region} \bigcup\limits_{d \in cComp} \{(r, DD_d(h_r(r(i))))\} \end{array} \tag{4.13}$$

où :

- général = {surface, orientation, élongation} ;
- région = {haut, bas, gauche, droit, centre} ;
- cComp = {teinte, saturation, intensité} ;

- $r(i)$ est une fonction qui extrait les pixels d'une partie r d'une image i.

Exemple 4.1. $\mathcal{D} = \{moyen, \ldots, standard\} \cup GroupeCouleur$
où $GroupeCouleur$ *est:*

$$\begin{pmatrix} rouge \\ \vdots \\ magenta \end{pmatrix} \bigotimes \begin{pmatrix} pale \\ \vdots \\ vive \end{pmatrix} \bigotimes \begin{pmatrix} noir \\ \vdots \\ blanc \end{pmatrix} \bigotimes \begin{pmatrix} haut \\ \vdots \\ bas \end{pmatrix}$$

4.1.4 Taille de l'espace des descriptions

Les variables linguistiques « surface », « orientation » et « élongation » peuvent produire des descriptions discrètes de 0, 1 ou 2 éléments dans une image. Cela est dû au fait que chaque trapèze peut chevaucher seulement son ou ses deux voisin(s) direct(s). Cependant, une discrimination plus fine peut mener à un degré k plus élevé de chevauchements. En outre, le niveau de chevauchement dépend du seuil correspondant : en général, plus le seuil est grand, moins le chevauchement se produit. Formellement, le nombre de possibilités est donné par :

$$P(\mathcal{D}_i) = 1 + A^{k_i}_{|\mathcal{D}_i|} \tag{4.14}$$

où 1 représente l'ensemble vide et A^m_n (le nombre de combinaisons de m éléments dans un ensemble n d'objets) donne le nombre de cas des valeurs floues au dessus du seuil choisi.

En revanche, les variables « teinte », « saturation », et « intensité » produisent un nombre exponentiel de cas. Par exemple, l'ensemble des teintes différentes qui apparaissent dans une image peut être un sous-ensemble de toutes les teintes possibles, pour une coupe α suffisamment basse. Par conséquent, le nombre de possibilités est donné par :

$$P(\mathcal{D}_j) = 2^{|\mathcal{D}_j|} \tag{4.15}$$

En conclusion, la taille de l'espace de concept associée à $\mathcal{D}$ est donnée par la formule suivante :

$$\prod_{i \in \{\text{surface,orientation,élongation}\}} P(\mathcal{D}_i) \times |\text{région}| \times \prod_{j \in \{\text{teinte,saturation,intensité}\}} P(\mathcal{D}_j) \tag{4.16}$$

La taille de l'espace de concept est $(1+5+4) \times (1+3+2) \times (1+5+4) \times 5 \times (2^7 \times 2^3 \times 2^4) = 10 \times 6 \times 10 \times 5 \times (128 \times 8 \times 16) = 49\,152\,000$. Cela est de loin au-delà de la taille de la plus grande base de données d'images qu'on peut imaginer, et donne une idée sur la puissance distinctive de cette technique.

Cependant, plusieurs images doivent être groupées et décrites par la même description, même pour de petites bases de données afin de réaliser une classification réelle. (Le risque de cette méthode est d'avoir un ensemble d'images partageant une propriété par paires. Cela produirait une explosion combinatoire, c'est-à-dire, un treillis avec un nombre exponentiel de nœuds.)

4.1.5 Modèles résultants

Dans cette section un modèle flou pour la recherche d'image, incluant les informations de couleur ainsi que les informations générales de forme (orientation, élongation, surface) a été décrit. Pour les informations de couleur, une division syntaxique de l'image en cinq parties (haut, bas, gauche, droite et centre) a été utilisée afin de séparer les objets réels d'une manière efficace. Un modèle binaire (semblable à un modèle basé sur des mots clés) a été dérivé, en appliquant un seuil au modèle flou. Ce modèle binaire servira à la construction du treillis de Galois que nous détaillerons à la section suivante.

4.2 Treillis de Galois et recherche d'information

La notion de treillis de Galois d'une relation (ou treillis de concepts) est à la base d'une famille de méthodes de classification conceptuelle. Introduite par Barbut et Monjardet [10], cette approche a été popularisée par Wille qui a utilisé la notion de treillis de Galois comme base de l'analyse formelle de concepts [126]. Wille propose de considérer chaque élément du treillis comme un concept formel et le graphe (diagramme de Hasse) comme une relation de généralisation / spécialisation entre les concepts. Le treillis est donc perçu comme une hiérarchie de concepts. Chaque concept est une paire composée d'une extension représentant un sous-ensemble des instances de l'application et d'une intension représentant les propriétés communes aux instances.

Plusieurs méthodes de classification conceptuelle basées sur les treillis de Galois ont été utilisées dans diverses applications [29, 131, 90] notamment pour la recherche documentaire où chaque concept du treillis généré à partir d'une relation d'indexation correspond à un ensemble de documents décrits par les termes d'index communs. Dans la perspective de la recherche booléenne, chaque concept peut être vu comme une requête formée de la

conjonction des termes d'index du concept (les éléments de son intension). Ainsi, en se basant sur la représentation de la collection de documents dans un treillis de Galois, la recherche documentaire bénéficie de la combinaison de deux modes d'interaction dans un même espace de recherche. En effet, le graphe représente une relation de généralisation / spécialisation entre les requêtes pouvant être satisfaites par les documents de la collection. La recherche est effectuée par une combinaison libre de :

1. la spécification directe de termes d'index, résultant en un saut dans le concept le plus général incorporant les termes spécifiés et les termes du concept de départ;
2. la navigation libre en suivant les arcs du graphe du treillis.

Le parcours d'un arc correspond à un élargissement (généralisation) ou un raffinement (spécialisation) minimal par rapport à la requête correspondant au concept courant.

Nous introduisons dans la suite, les notions formelles définissant les treillis de Galois et les algorithmes proposés pour les construire puis nous présenterons l'utilisation des treillis de Galois dans $Click^{Im}_{AGE}$ pour la navigation dans une base d'images.

4.2.1 Généralités sur les treillis de Galois

Nous présentons dans cette section les notions les plus usuelles relatives aux treillis de Galois. Ces notions sont rappelées dans divers travaux étudiant ou utilisant les treillis de Galois [30, 29, 126].

4.2.1.1 Treillis

Définition 4.8 (Treillis)**.** Un treillis est un graphe orienté, connexe, sans cycle et comportant un nœud supérieur (n'étant l'arrivée d'aucun arc) et un nœud inférieur (n'étant le point de départ d'aucun arc).

Une telle structure est généralement définie par une opération d'ordre partiel sur les nœuds, ainsi X est le père de Y équivaut à $Y \leq X$.

La structure de treillis est donc une structure abstraite très générale, qui peut par exemple servir à ranger les entiers naturels suivant leurs multiples. On définit alors la relation d'ordre partiel sur les entiers naturels comme suit :

$$\forall (X, Y) \in \mathbb{N}, X \leq Y \Leftrightarrow \exists Z \in \mathbb{N}, Y = X \times Z.$$

	a	b	c	d	e	f	g	h	i
1	1	0	1	0	0	1	0	1	0
2	1	0	1	0	0	0	1	0	1
3	1	0	0	1	0	0	1	0	1
4	0	1	1	0	0	1	0	1	0
5	0	1	0	0	1	0	1	0	0

Table 4.1 – Exemple de relation binaire entre des chiffres et des lettres [30]

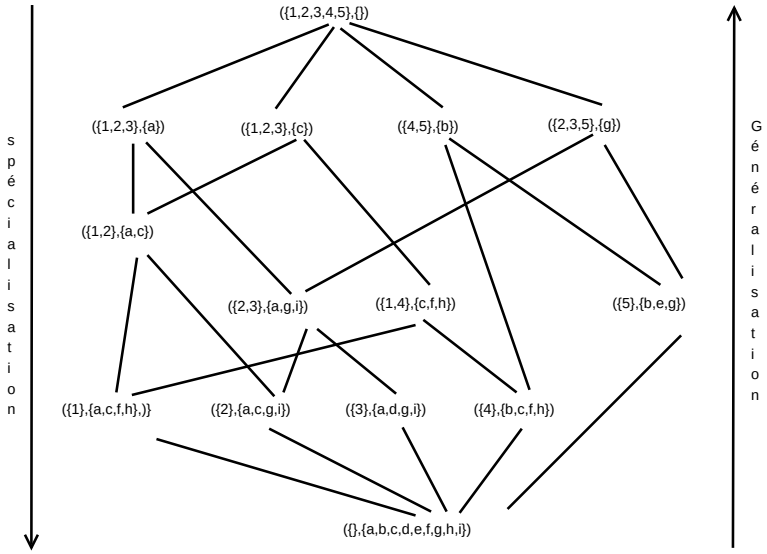

Figure 4.3 – Exemple de treillis de Galois [30]

4.2.1.2 Treillis de Galois

La structure qui nous intéresse est un cas particulier de treillis très couramment utilisé : *le treillis de Galois*. Celui-ci est utilisé pour représenter une relation binaire quelconque entre deux ensembles. Classiquement, on se base sur un ensemble d'objets et un ensemble de propriétés.

Plus formellement, si les deux ensembles sont notés O et A, O représentant les objets et A les attributs, on considère une relation binaire $R : O \times A \rightarrow [0, 1]$ telle que pour tout $(o, a) \in (O \times A)$, $o\mathcal{R}a$ signifie « o a la propriété a » [119].

Par exemple, on peut imaginer que O est un ensemble de documents et A un ensemble de mots-clés qui peuvent être associées ou non (de façon booléenne) à ces documents. Notons tout de même que dans le cas général, O et A jouent des rôles parfaitement identiques.

Chaque nœud du treillis de Galois est une paire, notée (X, X'), composée d'un en-

semble $X \subset O$ et d'un ensemble $X' \subset A$. Chaque paire doit être une *paire complète*, conformément à la définition ci-dessous.

Définition 1 (Paire complète)**.** *Une paire* (X, X') *est* complète *selon* $\mathcal{R}$ *si et seulement si les deux propriétés suivantes sont respectées :*

- $X' = f(X)$ *avec* $f(X) = \{x' \in A \mid \forall x \in X, x\mathcal{R}x'\}$
- $X = f'(X')$ *avec* $f'(X') = \{x \in O \mid \forall x' \in X', x'\mathcal{R}x\}$

Cela signifie que X' est l'ensemble des propriétés partagées par les éléments de X, et que réciproquement X est l'ensemble des instances comportant au moins les propriétés incluses dans X'.

On remarque que $f(\emptyset) = A$ et $f'(\emptyset) = O$. On verra par la suite que les paires $(\emptyset, A)$ et $(O, \emptyset)$ sont en général respectivement le nœud inférieur et le nœud supérieur du treillis. Si ce n'est pas le cas, c'est qu'une propriété est partagée par l'ensemble des instances ou qu'une instance possède toutes les propriétés.

De ces paires complètes, seules les paires maximales doivent être gardées. Cela signifie que s'il existe une instance $x \notin X$ qui possède toutes les propriétés de X', la paire (X, X') doit être étendue à $(X \cap \{x\}, X')$. Réciproquement, s'il existe une propriété extérieure à X' partagée par toutes les instances de X, la paire doit être étendue pour inclure cette propriété.

4.2.1.3 Algorithmes de construction des treillis de Galois

Il y a typiquement deux approches pour construire un treillis de Galois :

- l'approche « globale » ;
- l'approche « incrémentale ».

La première méthode consiste à construire tout le treillis en une exécution. Cela permet d'optimiser la construction en insérant les paires dans l'ordre le plus adéquat. Les algorithmes basés sur cette méthode sont donc plus efficaces en complexité que les algorithmes incrémentaux. Un algorithme de ce type a été proposé par [119].

La deuxième méthode, au contraire, vise à construire le treillis par insertion successive d'instances (en permettant également l'ajout de nouvelles propriétés lorsque cela est nécessaire). Cette approche se montre très adaptée lorsque les données ne sont pas toutes connues dès la mise en service du système, puisque l'ajout d'un seul élément sera plus efficace que la reconstruction complète du treillis [30].

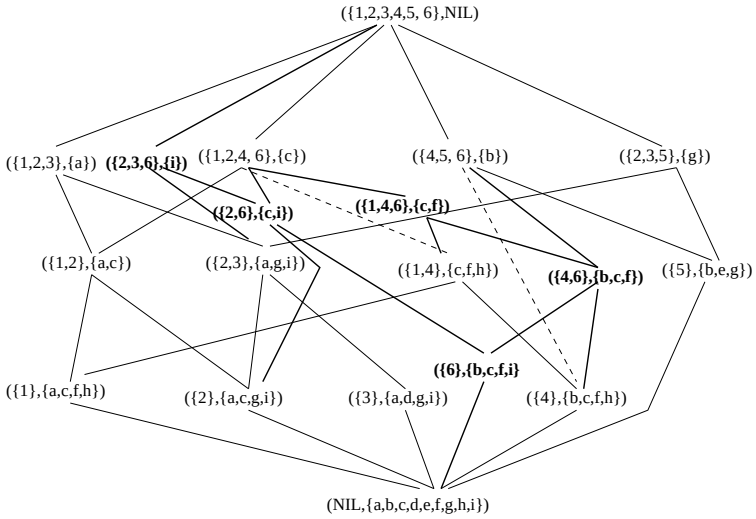

Figure 4.4 – Mise à jour d'un treillis de Galois [30]

4.2.1.4 Description de la mise à jour incrémentale

Le problème de la mise à jour incrémentale n'est pas simple. Par exemple, pour ajouter la paire $(\{6\}, \{b, c, f, i\})$ au treillis illustré en figure 4.3, il faut faire toutes les modifications montrées en figure 4.4 [30]

En effet, pour ajouter une paire à un treillis de Galois il ne suffit pas de trouver quels doivent être ses pères et quels doivent être ses fils : il faut souvent créer de nouvelles paires.

Considérons l'insertion de l'instance $x*$. Cela équivaut à insérer dans le treillis la paire $(x*, f(x*))$. On note G le treillis initial et $G*$ le treillis final.

Pour chaque paire (X, X') telle que $X \subset f(x*)$, X doit être « augmenté » de $x*$. Ces paires sont nommées *paires modifiées* et les autres *anciennes paires*. De plus de *nouvelles paires* vont être créées.

Ces nouvelles paires seront de la forme $(Y \cup \{x*\}, Y' \cap f(\{x*\}))$, où (Y, Y') est une paire de G. La paire (Y, Y') est appelée *g*énératrice de la nouvelle paire.

4.3 Treillis de Galois et navigation dans une base d'images

Comme nous l'avons souligné, les systèmes de recherche d'information multimédia par navigation sont généralement basés sur l'exploration d'une structure d'arbre ou de graphe de voisinage. Un problème important de cette approche est la difficulté de maintenir la

classification. Un autre problème est la rigidité de la classification où un seul chemin d'accès est possible pour retrouver une information à partir d'un point d'entrée unique. Et aussi une mauvaise décision dans le parcours conduit à des résultats inattendus.

L'utilisation de treillis de Galois généré à partir d'une relation d'indexation est une alternative attrayante. Le treillis de Galois se construit sur une classification préalable des images permettant de faire des regroupements par similarité se basant sur une solide connaissance du domaine mathématique et informatique.

4.3.1 La classification automatique des images

La classification automatique d'images est une de ces techniques, fruit des recherches récentes. Elle fournit un cadre conceptuel pour permettre le regroupement en classes d'images sur lesquelles on aura rassemblé des informations sur les caractéristiques communes à ces images. En quelque sorte c'est une méthode mathématique qui propose automatiquement des regroupements rationnels.

La base du raisonnement qui justifie ce regroupement n'est pas purement logique mais comme une forme d'apprentissage inductif. Vis à vis de tel ou tel critère, des images ayant certaines similarités se rassemblent et forment une classe. Cette aptitude dans le processus laisse deviner toute la complexité et la difficulté de sa mise en œuvre principalement à cause du grand nombre de variables d'entrées qui servent à décrire les images.

En effet, en présence d'un grand nombre de variables, les méthodes d'apprentissage souffrent la plupart du temps d'une grande variance qui dégrade leur précision et de plus elles présentent des temps de calcul très élevés. Pour gérer ce problème de dimensionnalité, la classification d'images repose généralement sur un pré-traitement spécifique à chaque problème qui réduit sa complexité en extrayant des caractéristiques pertinentes. Celles-ci sont ensuite utilisées en entrée d'une méthode traditionelle d'apprentissage automatique éventuellement ajustées pour l'application.

Cependant, les avancées en apprentissage automatique ont fait apparaître des méthodes capables de traiter des problèmes de plus en plus complexes sans utiliser aucune information a priori sur le domaine d'application. Le treillis de Galois est une de ces méthodes.

Dans cette section nous montrerons comment $Click^{Im}_{AGE}$ utilise les treillis de Galois comme une méthode générique pour la classification d'images en vue de la navigation.

Comme nous l'avons souligné un treillis de Galois se construit à partir d'une relation binaire portant sur des domaines discrets. Dans notre cas, le contexte général ou contexte global $\mathcal{K}$ est le contexte relatif à l'ensemble des images $\mathcal{I}$, auquel on associe un ensemble $\mathcal{D}$ des descriptions (issues des métadonnées qui leur ont été associées), ainsi que la relation

binaire :

$$R : \mathcal{I} \times \mathcal{D} \tag{4.17}$$

Où, $\mathcal{I}$ est l'ensemble des images et $\mathcal{D}$ celui des descriptions *discrétisées* issues des métadonnées qui leur ont été associées (la discrétisation est essentielle car elle détermine l'organisation des données pour la construction du treillis de Galois).

Informellement, il s'agit d'une transformation de la relation qui construit :

- un graphe orienté sans cycle,
- dont les nœuds sont caractérisés par un un ensemble d'images, c'est-à-dire une *extension* (incluse dans $\mathcal{I}$) auquel on associe un ensemble de descriptions, c'est-à-dire une *intension* (incluse dans $\mathcal{D}$),
- possédant un unique nœud $sup = (\mathcal{I}, \emptyset)$ sans arcs entrants et un unique sommet $inf = (\emptyset, \mathcal{D})$ sans arcs sortants,
- pour lequel chaque paire de sommets (I, D) et (D', I') a un unique nœud supérieur en commun, $(I \cup I', D \cap D')$, ainsi qu'un unique nœud commun inférieur, $(I \cap I', D \cup D')$.

Les nœuds du treillis sont donc des paires de la forme (X, X') où $X \subset \mathcal{I}$ et $X' \subset \mathcal{D}$. $\mathcal{C} = \mathcal{I} \times \mathcal{D}$ est l'ensemble des nœuds du treillis. Ces noeuds sont des paires complètes, c'est-à-dire que :

- $X = \{i \in \mathcal{I} | \forall d \in X', (i, d) \in R\}$.
- $X' = \{d \in \mathcal{D} | \forall i \in X, (i, d) \in R\}$;

On a vu que de ces paires complètes, seules les paires maximales doivent être gardées. Cela signifie que toutes les images $i \in X$ sont décrites par tous les descripteurs $d \in X'$ et tous les descripteurs $d \in X'$ sont associés à toutes les images $i \in X$.

À partir de cette définition, on peut définir l'ordre partiel suivant :

$$\forall (C_1 = (X_1, X_1'), C_2 = (X_2, X_2')) \in \mathcal{C}^2, C_1 < C_2 \Longleftrightarrow X_1 \subset X_2 \Longleftrightarrow X_2' \subset X_1'$$

Cet ordre partiel est utilisé pour générer le diagramme de Hasse du treillis de Galois comme suit : une paire (C_1, C_2) est créée s'il existe $C_1 < C_2$ et il n'existe pas une autre paire C_3 tel que $C_1 < C3 < C_2$.

Le treillis de Galois ainsi créé représente donc une relation de généralisation / spécialisation. Son nœud *sup* contient toutes les images auxquelles aucune propriété n'est associée. Les autres nœuds du treillis sont des regroupements d'images ayant des propriétés communes. Le parcours du treillis dans le sens descendant consiste à une spécialisation

de propriétés et le parcours dans le sens ascendant à une généralisation de propriétés. La section 4.3.2 détaille le principe de la navigation dans de tel treillis.

Cette structure générale peut d'une part être utilisée en vue d'obtenir une classification des images sans prendre en compte des informations de nature contextuelle. On peut ainsi classer les images en fonction d'informations générales telles que les images (intérieur/extérieur), les images (jour/nuit), etc. D'autre part, elle peut aussi être utilisée pour l'indexation d'un type bien particulier d'image, comme par exemple les images médicales, les images satellitaires, etc. Dans ce cas, l'indexation est contextuelle car basée sur une problématique précise, et le résultat obtenu répond aux attentes des utilisateurs. Ces systèmes spécifiques, même si leur utilisation est limitée à un type d'image, permettent cependant de répondre à de nombreuses demandes de la part d'utilisateurs de systèmes d'indexation d'images.

Pour notre part, le treillis de Galois nous pemet de bien répondre à la problématique que nous avons posé à savoir le développement d'un système générique de recherche des vidéos et images du patrimoine culturel marocain. Ainsi, les propriétés retenues pour la description d'une image peuvent variées d'une application à une autre. Elles peuvent être liées au contenu intrinsèque des images comme la couleur, ou au contraire peuvent ajouter des sémantiques au contenu à travers des mots-clés.

Le choix d'une technique de description consistera donc à dresser une liste de métadonnées sur les images. Sachant que l'extraction de descripteurs permettant de traduire au mieux le contenu d'une image est une tâche difficile, il est important de considérer à la fois plusieurs descripteurs sinon les nœuds du treillis seront formés par des classes générales d'images (par exemple juste des images au niveau de gris) ce qui donne un nombre élevé de réponses au niveau d'un nœud. Ensuite il faudra proposer des techniques de comparaison entre ces propriétés, basées sur des distances de préférence, mais devant être en définitive des similarités perceptuellement significatives pour l'œil humain.

4.3.2 Le Principe de la navigation

La définition du principe de navigation sur un treillis de Galois est un aspect important permettant d'offrir un moyen de consultation efficace.

En fait, le treillis de Galois peut être vu comme un espace de recherche dans lequel on évolue en fonction des descriptions validées. La navigation est effectuée dans le diagramme de Hasse, dont les nœuds sont constitués d'images ayant des propriétés communes. Les nœuds sont reliés à une multitude d'autres noeuds par des liens. Le parcours du diagramme consiste soit à une spécialisation soit à une généralisation de la requête d'un utilisateur.

Il s'agit donc à partir d'un concept de progresser étape par étape au sein du treillis de Galois par validation de nouvelles caractéristiques et par conséquent réduction de l'ensemble d'images, jusqu'à ce que l'utilisateur soit satisfait ou qu'il décide que la base ne contient pas l'information qu'il cherche.

Le tableau 4.2 donne un exemple de relation binaire définie sur 5 images, décrites par trois propriétés : paysage, personnage, bleu.

	paysage	personne	bleu
i_1	1	1	0
i_2	0	1	0
i_3	1	0	1
i_4	0	0	0
i_5	0	0	0

Table 4.2 – Exemple de relation binaire entre des images et leur propriété

La figure 4.5 donne la représentation graphique de cet exemple. Il s'agit d'un treillis d'inclusion.

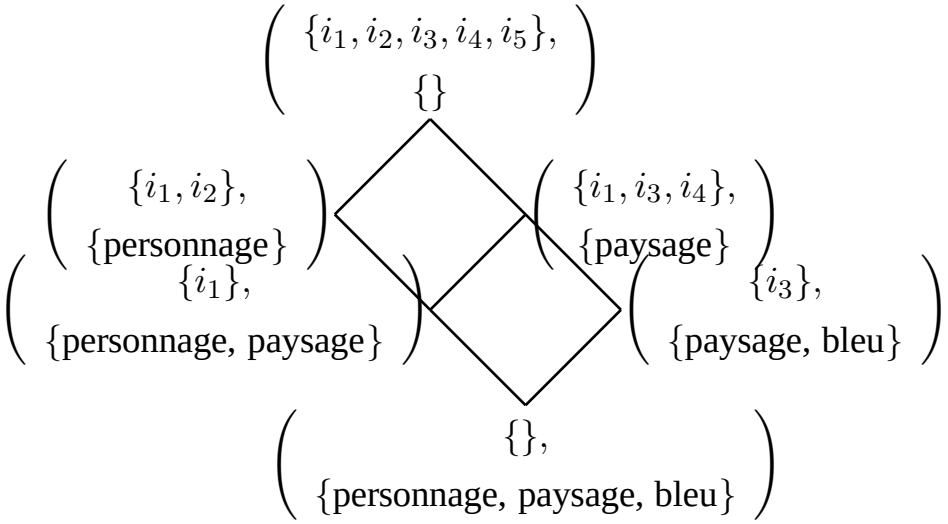

Figure 4.5 – Un exemple de treillis de Galois

La description d'une image est traduite ici par une chaîne de caractères : personnage, paysage, bleu permettant de décrire une image avec un personnage dans un paysage avec un ciel bleu. L'utilisateur peut, en descendant de proche en proche, s'intéresser à des classes d'images de plus en plus précises ou, au contraire, relâcher des contraintes en suivant des arcs ascendants. Le treillis de Galois ainsi créé permet à un utilisateur d'affiner sa requête

au moyen d'une navigation.

La technique ainsi définie offre de nombreux avantages pour la navigation. La classification des images et la construction du treillis sont effectuées hors ligne, les temps de réponse aux requêtes sont donc optimaux. Elle permet des recherches conjonctives sur l'ensemble des métadonnées élicitées, en ne privilégiant aucun point de vue. De plus, les recherches sont dichotomiques en le nombre de propriétés retenues. Enfin, de par ces nombreuses interconnexions, il évite l'écueil de l'arborescence où chaque « erreur » dans le choix d'un nœud fils nécessite des retours en arrière pour descendre dans une nouvelle branche.

4.3.3 La visualisation du treillis

Une fois les images classifiées et le treillis construit, il est nécessaire de représenter les informations obtenues de façon à les rendre facilement accessibles par l'utilisateur. L'interface de visualisation est donc la finalisation de la conception du système de navigation. Elle doit rendre la navigation dans le treillis la plus aisée possible, mais aussi donner beaucoup d'information à la fois sans pour autant encombrer les information rendues.

Il n'est pas convivial de visualiser directement toute la structure du treillis de Galois dans le processus de navigation. C'est pourquoi nous adoptons l'approche qui consiste à montrer les prédécesseurs et successeurs immédiats par rapport au contexte courant de navigation. Cette technique évite non seulement la surcharge de l'écran de navigation, mais aussi la désorientation de l'utilisateur. Il lui est possible à tout moment de savoir d'où il vient (prédécesseurs), où il est (courants), et où il va (successeurs). Le chapitre 7 détaille notre approche pour la visualisation du treillis.

4.4 Conclusion

Dans ce chapitre, nous avons présenté le système $Click^{Im}_{AGE}$. Nous avons vu que pour $Click^{Im}_{AGE}$, la représentation des données est un ensemble de métriques basées sur le contenu des images. Ces métriques sont principalement des informations sur la forme générale de l'image (taille, orientation, élongation) et des informations de couleur classées par zones : les couleurs sont issues d'une segmentation de l'espace à partir du repère HSV, ces informations de couleur sont associées aux images en utilisant la logique floue.

Nous avons aussi présenté les treillis de Galois dont la répresentation graphique est une structure d'arbre qui a été utilisée par beaucoup d'auteurs pour la recherche d'information.

Enfin, pour pouvoir utiliser les treillis de Galois pour la navigation dans une base

d'image, nous avons vu que dans $Click^{Im}_{AGE}$, les degrés d'appartenance flous sont supprimés et une relation binaire entre images et métadonnées est produite.

Dans le chapitre suivant, nous présenterons notre proposition pour la modélisation de la vidéo, ainsi que la liaison entre notre modèle et $Click^{Im}_{AGE}$. Le modèle resultant permettra de traiter les vidéos comme une extension qui s'appuie sur la modélisation des images de manière quasi transparente.

CHAPITRE 5

MODÉLISATION DE LA VIDÉO

L'étude des systèmes de recherche d'information dans la partie I nous permet d'affirmer que le développement d'un système de recherche d'information (SRI) multimédia obéit aux mêmes étapes principales que celles d'un SRI textuel, à savoir (i) la modélisation des documents, (ii) la traduction des requêtes, (iii) la mise en correspondance de ces dernières avec les descriptions des documents et (iv) éventuellement une phase de rétroaction pour améliorer la qualité de la réponse qui est nécessairement approximative car basée sur des calculs de distances ou (dis)similarités dans la plupart des cas [27] (cf. figure 5.1) .

Ce chapitre présente notre proposition pour la modélisation des médias visuels. Nous y proposons une (méta)modélisation de la vidéo. Cette proposition fait ressortir les deux aspects évoqués dans le chapitre 2 à savoir la structuration d'une vidéo et son annotation. Le modèle que nous proposons est suffisamment général bien que souffrant de quelques limitations pratiques dues au choix d'un SGBD relationnel pour l'implémentation.

Une vidéo pouvant être perçue comme une succession d'images fixes, nous proposons de modéliser les vidéos comme une extension qui s'appuie sur la modélisation des images de manière quasi transparente.

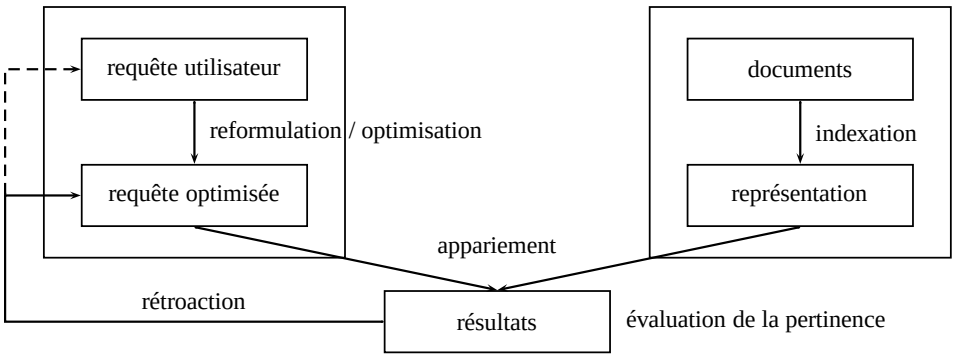

Figure 5.1 – Principe général de la recherche d'information

Les images, sont décrites et indexées *via* les mécanismes mis en œuvre par $Click^{Im}_{AGE}$.

La liaison entre la structure et les métadonnées est décrite. La liaison avec la base d'images est également évoquée.

Le principal but poursuivi est, d'une part, de répondre aux besoins de *généricité* et de *flexibilité* exprimés dans le chapitre 1 et, d'autre part, de proposer un système qui permette de naviguer en basculant indistinctement entre images et vidéos.

5.1 Modélisation des vidéos

La modélisation d'un document vidéo est l'étape pendant laquelle le document se voit conférer un statut conceptuel dans la base gérée par un système de recherche. Comme nous l'avons souligné, notre travail porte sur la conservation du patrimoine culturel marocain filmé et photographié. Notre objectif est de produire un système qui permette de naviguer en basculant indistinctement entre images et vidéos. Nous avons donc à gérer un SGBD de vidéos et d'images. Les vidéos ont un rôle d'animation alors que les images fixes permettent de mieux appréhender les objets artisanaux, entre autres, ces dernières étant de meilleure qualité visuelle et de plus grande taille que les vidéos.

Dans le chapitre précédent, nous avons présenté $Click^{Im}_{AGE}$ qui modélise les images en se basant sur leur contenu. Dans cette section, nous présentons notre proposition dans l'élaboration du schéma de base de données de la partie vidéo du SGBD.

En effet, les grandes étapes de la création d'un SGBD vidéo sont :

1. la définition d'un schéma suffisamment général pour couvrir sinon tous les besoins du moins une large gamme ;
2. l'alimentation de la base avec des – descriptions de – vidéos ;
3. l'interrogation, au sens large, de la base en exploitant tout à la fois des informations sémantiques et des informations sur le contenu même des vidéos.

Pour ce qui est de la définition d'un schéma, une vidéo n'est pas une simple suite d'images. Elle illustre aussi un scénario, elle possède un langage avec sa logique et sa grammaire ainsi que des techniques précises de montage [1]. Lors de l'indexation, il faut utiliser, voire redécouvrir ces informations, briser le caractère continu d'une vidéo et aboutir à une structuration de celle-ci en segments sémantiques, dans le meilleur des cas, ou basés sur le contenu, à défaut. De plus, les règles de composition d'une vidéo varient suivant la nature de la vidéo (film, documentaire, clip, publicité, etc.). Enfin, les opérations de montage et de

[1] Chacune des nombreuses techniques de montage vise à produire une émotion particulière chez le spectateur.

manipulation numérique de la vidéo engendrent de nouvelles techniques d'écriture visuelle, difficiles à prendre en compte dans leur totalité [2].

Notre proposition (cf. figure 5.2) tente d'effectuer une synthèse, difficile, des propositions existantes. Il peut être vu tout à la fois comme un guide dans l'élaboration d'un modèle particulier et comme un modèle suffisamment générique pour pouvoir supporter un grand nombre de métadonnées associées à des vidéos variables.

Un point fort de notre modèle est sa flexibilité. Cette flexibilité se traduit, d'une part, par une décomposition hiérarchique paramétrable des types de vidéos et, d'autre part, par la variété des descripteurs que l'on peut associer à chaque décomposition d'une vidéo.

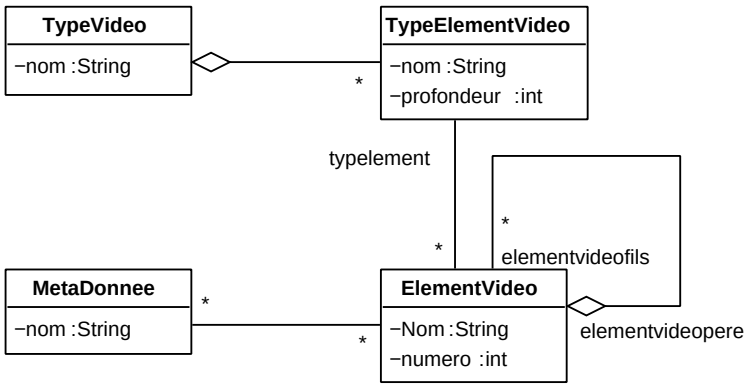

Figure 5.2 – (Méta)Modèle d'indexation de la vidéo dans le formalisme UML

5.1.1 Décomposition d'une vidéo

Un système capable de s'adapter à différents types de vidéos doit autoriser une décomposition variable. En fait, il s'agit d'un métamodèle. Il sera instancié dynamiquement en fonction des besoins et ses instances stockées dans le SGBD. Il permet d'associer des structurations de profondeur variable aux documents vidéo. Les noms donnés aux différents éléments permettent aussi de fournir une sémantique simple (pour sélectionner les fictions, les documentaires, etc.) où les passages scéniques (épisodes, séquences, etc.) qui, dans la plupart des modèles vus au chapitre 2, sont fixés *a priori*. Cela se traduit par l'unique classe *ElémentVidéo* munie d'une agrégation récursive.

[2] Par exemple, dans « L'affaire Thomas Crown », avec Steve McQueen et Faye Dunaway, certaines parties du film présentent des actions parallèles dans le temps sous la forme d'un patchwork d'incrustations. Classiquement, les actions parallèles sont présentées soit à la suite l'une de l'autre, soit plus souvent en découpant les deux scènes et en alternant leur passage (montages alterné ou parallèle). Dans le cas de ce film, il est impossible de procéder automatiquement à un découpage en plans vraiment significatifs.

Le fait qu'il n'y ait qu'une seule classe évite une décomposition minimale [103] ou jugée *a priori* minimale [116] : vidéo composée de plans. Une vidéo (très) courte peut n'être composée que d'un seul plan, appelé plan-séquence dans le langage cinématographique. De plus, et en toute logique, la feuille de décomposition d'une vidéo est l'image [17, 26]. Bien sûr, il est hors de question de vouloir décomposer chaque plan en autant d'images. En revanche, une ou quelques images représentatives du plan peuvent se situer à cet endroit-là [17]. À l'opposé, nous ne supposons pas que la racine de toute description est une vidéo. Par exemple, les films, les épisodes d'un feuilleton télévisé aussi bien que des documentaires peuvent être groupés au sein de séries comme (Hout al Bar, Lalla Fatima, Moul Taxi, ...). Des descriptions communes pourront alors être associées aux séries, évitant ainsi d'avoir à les répéter pour chaque vidéo.

5.1.2 « Typage » des vidéos

La classe unique *ElémentVidéo* est accompagnée des métadonnées qui permettent son interprétation (et non son indexation). Ce sont les classes *TypeElémentVidéo* et *TypeVidéo*. La traduction en vue d'une implémentation relationnelle des classes qui décrivent la structuration variable des vidéos est donnée en figure 5.3 [3].

Pour la relation *ElémentsVidéos* correspondant à la classe *ElémentVidéo*, le couple (*IdPère*, *Rang*) est une clé candidate.

L'attribut *Rang* traduit le fait que les éléments vidéo fils sont ordonnés dans l'agrégation récursive :

- le père et le fils appartiennent au même type de vidéo ;
- la différence de profondeur entre le père et le fils est exactement de 1.

L'exploitation d'un métamodèle entraîne toujours des surcharges à l'exécution à cause de la dynamique nécessaire à son interprétation. Cet inconvénient doit être mis en balance avec la facilité d'évolution du schéma.

L'alternative serait d'utiliser le schéma de la figure 5.3 comme un patron. Créer un nouveau type de vidéo se traduit par la création d'autant de relations que de niveaux. Le métaschéma reste partiellement accessible *via* le catalogue du SGBD, l'occupation mémoire est réduite, la rédaction des requêtes SQL et leur exécution est plus rapide puisque portant sur des relations nommées [4] et plus nombreuses, donc de cardinalités plus faibles.

[3] Le nom de la classe est au singulier car il s'agit d'un « moule » décrivant les propriétés de chaque intance. Le nom de la relation est au pluriel car il s'agit, formellement, d'un *ensemble d'instances*.

[4] Dans la version précédente, les relations nommées peuvent être obtenues grâce à des vues, éventuellement matérialisées.

ElémentsVidéos
IdElémentVidéo : identifiant
IdPère : identifiant **référence** ElémentsVidéos
Rang : entier
IdType : identifiant **référence** TypesElémentsVidéos

TypesElémentsVidéos
IdTypeElémentVidéo : identifiant
Profondeur : identifiant
Nom : chaîne
IdTypeVidéo : identifiant **référence** TypesVidéos

TypesVidéos
IdTypeVidéo : identifiant
Nom : chaîne

Figure 5.3 – Sous-schéma relationnel correspondant à la description de la vidéo de la figure 5.2

Enfin, nous ne pouvons pas clore cette section sans mentionner que, au-delà d'une hiérarchie extensible, c'est une véritable *grammaire extensible* qu'il faudrait offrir. Autrement dit, les niveaux devraient être variables non seulement entre types de vidéos mais au sein d'une même instance de vidéo. De plus, les différents fils d'un nœud devraient pouvoir porter des libellés différents. Par exemple, un journal télévisé se compose de génériques de début et de fin, débute par une énumération des titres développés, comporte une partie actualité avec des nouvelles courtes (seulement annoncées par le présentateur), des nouvelles donnant lieu à un reportage et des nouvelles longues se traduisant par plusieurs reportages et/ou entretiens, suivies éventuellement d'une partie culturelle et s'achevant, quelques années auparavant, par les prévisions météorologiques.

Une telle modélisation reste réalisable dans le cadre d'un SGBD relationnel mais elle devient très complexe. Il faut prendre en compte l'ordre des sous-éléments vidéos, leur nombre d'occurrences, les répétitions de groupes, les alternatives et éventuellement les imbrications récursives. Il vaut mieux attendre la maturité des SGBD XML natifs pour s'engager dans une telle voie.

Soulignons toutefois que, dans les deux cas, le *principe* est le même. La structure de la vidéo est capturée par une décomposition hiérarchique, plus ou moins complexe, plus ou

moins souple.

5.1.3 Extensibilité des métadonnées

Les métadonnées sont éminemment variables. Nous avons vu qu'un film dans un vidéo-club peut être décrit par une jaquette ; à tous les éléments de la vidéo peuvent être associés des strates ou des résumés; les informations sur le contenu se retrouvent essentiellement dans les niveaux les plus bas, plans et scènes, etc.

Dans le schéma de la figure 5.2, la classe *MétaDonnée* n'est que la *racine d'un graphe d'héritage*. Il est ainsi possible d'étendre le système afin de prendre en compte de nouveaux besoins et de nouveaux types de vidéos.

La figure 5.4 donne la traduction de la classe dans le modèle relationnel et présente quelques métadonnées particulières. Le passage du modèle à objets au modèle relationnel se fait selon l'une des trois techniques classiques [5]. Nous avons opté pour la traduction qui consiste à stocker dans la relation racine *tous* les identifiants de métadonnées, *accompagnés* du nom de la relation où l'on peut trouver les informations effectives.

Cela dit, à partir des travaux référencés, il est possible de faire émerger une base d'utilisation assez large qui serait une extension de la notion de strate de manière à englober des objets visuels et/ou sémantiques en permettant :

- une classification des objets ;
- une extensibilité de cette classification ;
- des attributs multiples ;
- des informations temporelles ;
- des informations spatiales à l'intérieur des intervalles temporels.

Le schéma de la figure 5.5 représente le schéma « fédérateur » correspondant. Les strates forment un cas particulier où chaque objet est décrit uniquement par une seule paire attribut / valeur. Une jaquette est traduisible par un objet auquel on associe une liste de paires. Par rapport au schéma de la figure 2.5, il est possible d'insérer plusieurs paires dont l'attribut est de type « acteur ». La classification des types d'attributs remplace facilement l'héritage [36, 3].

Pour contrôler des contraintes de cardinalité, de nécessité, etc., il faudrait prévoir le métamodèle correspondant. L'extension à d'autres types d'associations que la spécialisation

[5] Les trois techniques sont les suivantes : une unique relation regroupant tous les attributs de toutes les sous-classes, et donc toutes les instances ; autant de relations que de sous-classes concrètes, réalisant une partition des instances ; et autant de relations que de classes, les attributs des instances d'une classe étant répartis entre les différentes relations correspondant à la classe d'appartenance et aux classes ancêtres.

MétaDonnées
IdMétaDonnée : identifiant
TypeMétaDonnée : { Jacquette, Strate, . . . }

Jaquettes
IdJaquette : identifiant **référence** MétaDonnées
Titre : chaîne
Réalisateur : chaîne
. . .
IdElémentVidéo : identifiant **référence** ElémentsVidéos

Transitions
IdTransition : identifiant **référence** MétaDonnées
TypeTransition : { FonduEnchaîné, . . . }
IdElémentVidéoPrécédant : identifiant **référence** ElémentsVidéos
IdElémentVidéoSuivant : identifiant **référence** ElémentsVidéos

Strates
IdStrate : identifiant **référence** MétaDonnées
TypeStrate : chaîne
ValeurStrate : chaîne

StratesElémentsVidéos
IdStrate : identifiant **référence** Strates
IdElémentVidéo : identifiant **référence** ElémentsVidéos

Figure 5.4 – Sous-schéma relationnel correspondant à la description des métadonnées de la figure 5.2 et quelques « sous-classes »

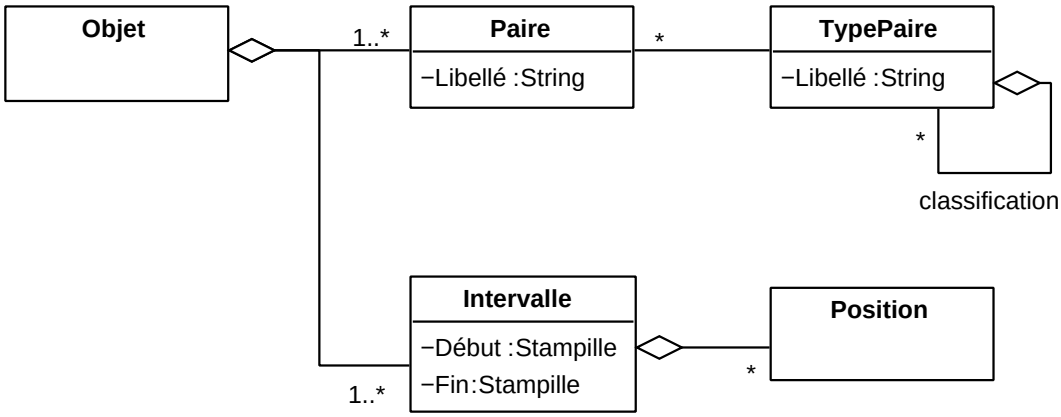

Figure 5.5 – Métadonnées génériques

nous entraîne progressivement vers une véritable *ontologie*. Par exemple, Hacid, Decleir et Kouloumdjian [32] prennent en compte les relations *entre* objets. Par rapport au schéma de la figure 5.5, cela reviendrait *généralement* à créer une sous-classe *Groupe* de la classe *Objet*, les attributs d'un groupe ayant la possibilité de référencer des objets et pas seulement une constante alphanumérique. D'autres propositions ont déjà été évoquées en fin de section 2.2.1.2.

5.1.4 Contrôles des associations entre vidéos et métadonnées

Il est indispensable de contrôler l'association de la figure 5.2 entre les éléments d'une vidéo et les métadonnées. En effet, on ne doit pas associer une jaquette à un plan ou, inversement, une transition à un film.

Le formalisme UML ne permet pas de traduire graphiquement une contrainte qui porte, d'une part, sur les *instances* de la classe *TypeElémentVidéo* et, d'autre part, sur les *sous-classes* de *MétaDonnée*.

L'association de la figure 5.2 est donc « seulement » l'union de diverses associations qui peuvent exister entre métadonnées et éléments vidéo. Notons que, contrairement à la proposition de Marcus et Subrahmanian [62], une métadonnée peut être associée à plusieurs « états » (cf. section 2.2.1.2).

Dans les deux sections suivantes, nous exposons comment cette association est matérialisée et contrôlée.

5.1.5 Associations entre vidéos et métadonnées

Cette association peut être particularisée sur des exemples de métadonnées. Trois cas sont illustrés sur la figure 5.4 :

1. Une jaquette fait référence à une seule vidéo (plus exactement un élément de vidéo qui doit être une vidéo cinématographique). Il s'agit d'une association entre un cas particulier de métadonnées et un cas particulier d'éléments vidéo. Les cardinalités de ses deux rôles sont 1..1-0..1 [6].
2. Une transition lente fait explicitement référence aux deux plans qu'elle sépare. Il faudrait également tenir compte des transitions lentes associées au début ou à la fin d'un seul plan, notamment les fondus au blanc ou au noir. Notez que dans le modèle de Thuong [114], une transition n'est toujours associée qu'à un seul plan. Dans le cas général, les cardinalités sont donc 1..2-0..2.
3. Enfin, il est intéressant de ne pas dupliquer une strate dans chaque plan, scène, etc., où elle s'applique. Une relation de jointure (*StratesElémentsVidéos*), traduisant une réelle association multivaluée de part et d'autre, doit alors être introduite. Ses cardinalités sont 1..*-0..*.

L'association de la figure 5.2 entre métadonnées et éléments de la vidéo peut être spécialisée ou générale.

Sur la figure 5.4, c'est la première approche qui a été choisie. En effet, les deux premières métadonnées (*Jaquette* et *Transition*) intègrent les associations en tant qu'attributs (*IdElémentVidéo*, *IdElémentVidéoPrécédant* et *IdElémentVidéoSuivant*). Il est possible de supprimer ces attributs en créant une relation *VidéosElémentsVidéos* (plus générale que *StratesElémentsVidéos* mais possédant le même schéma de relation).

Alternativement, il est possible de traduire l'union des associations sous la forme d'une vue, éventuellement matérialisée. À chaque modification de la hiérarchie de métadonnées, la définition de cette vue devra être elle-même modifiée. En tout état de cause, l'association entre métadonnées et éléments de la vidéo peut être vue sous une forme générique.

5.1.6 Contrôle des associations

Avec l'une ou l'autre approche, le SGBD n'est pas capable de contrôler les associations entre métadonnées et éléments particuliers d'une vidéo. La relation *ContraintesVidéosMétaDonnées* de la figure 5.6 explicite cette contrainte grâce à l'utilisation de l'attri-

[6] La séparation en deux relations ne permet pas de contrôler automatiquement la cardinalité maximale du rôle *Jaquette* dans la relation *ElémentVidéo*.

ContraintesVidéosMétaDonnées
IdTypeElémentVidéo : identifiant **référence** TypesElémentsVidéos
TypeMétaDonnée : { Jaquette, Strate, . . . }

Figure 5.6 – Relation explicitant les contraintes d'associations entre éléments de vidéos et métadonnées

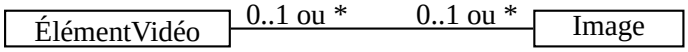

Figure 5.7 – Liaison entre les schémas de vidéos et d'images

but *TypeMétaDonnée* introduit dans le processus de traduction du modèle à objets vers le modèle relationnel (cf. classe *MétaDonnée* de la figure 5.4). Ainsi, cette relation contient l'ensemble des paires éléments vidéo / métadonnées autorisées. Soulignons que l'attribut *IdTypeElémentVidéo* se réfère à un élément vidéo *dans une vidéo particulière*. De la sorte, il est possible de différencier les associations autorisées entre les scènes d'un film et les scènes d'un documentaire, par exemple.

En s'appuyant sur la relation de jointure ou la vue *VidéoElémentVidéo*, il serait possible d'ajouter les cardinalités minimales et maximales de chaque rôle dans la relation *ContraintesVidéosMétaDonnées*.

5.1.7 Prise en compte de l'image

Comme nous l'avons souligné, les éléments indivisibles de la vidéo, c'est-à-dire les images, seront décrits et indexés *via* les mécanismes mis en œuvre par $Click^{Im}_{AGE}$.

Comme nous nous intéressons aux images et aux vidéos, il n'est pas possible d'intégrer systématiquement les premières comme niveau de décomposition le plus fin des secondes.

La base va donc être constituée de deux schémas de données multimédias *a priori* distincts, comme cela est proposé par Marcus et Subrahmanian [62]. (Les images sont elles-mêmes des données complexes, munies de nombreuses métadonnées et qui peuvent obéir à la modélisation décrite en section 2.2.1.2.) Mais il est inacceptable de dupliquer des sous-schémas de métadonnées communs à la sous-base d'images et à la partie qui traite les images représentatives dans la sous-base des vidéos.

La liaison entre les deux schémas s'effectue entre la classe *Image*, provenant du schéma de la sous-base d'images, et la classe *ElémentVidéo* du schéma de la sous-base de vidéos (cf. figure 5.7). *A priori*, cette dernière peut contenir des références à des images – représentatives – (ce qui peut être contrôlé *via* la classe *TypeElémentVidéo*).

Dans le cas d'une image qui provient d'une vidéo, l'identifiant de cet élément vidéo particulier est *aussi* l'identifiant de l'image correspondante dans le schéma de la sous-base d'images. Ce choix évite une relation de jointure pénalisante. Il évite aussi de modifier le schéma de la sous-base des images en introduisant une référence de ces dernières vers la sous-base des vidéos. La liaison reste donc *implicite*, quoique bien réelle.

Les fonctions 5.1 à 5.5 illustrent mieux notre méthode de modélisation de la vidéo ainsi que la liaison avec la base d'images :

$$\mathcal{V} \rightarrow \mathcal{V}_T \tag{5.1}$$

$$\mathcal{V}_T \times \mathbb{N} \rightarrow \mathcal{V}_{TE} \tag{5.2}$$

$$\mathcal{V}_E \rightarrow \mathcal{V}_{TE} \tag{5.3}$$

$$\mathcal{V}_E \not\rightarrow \mathcal{V}_E \tag{5.4}$$

$$\mathcal{V}_{TE} \times \mathcal{I} \tag{5.5}$$

Chaque vidéo $v \in \mathcal{V}$ est associée à un certain type de vidéo $\mathcal{V}_T$. Chaque type de vidéo obéit à une structure hiérarchique parfaitement balancée dont le nombre de niveaux est fixé par un entier n_T. Chaque niveau de cette hiérarchie est typé, $\mathcal{V}_T \times \mathbb{N} \rightarrow \mathcal{V}_{TE}$, où le niveau le plus bas de $\mathcal{V}_{TE}$ est toujours le « plan » ici. Les images clés sont associées aux plans, $\mathcal{V}_{TE} \times \mathcal{I}$, et traitées comme les images fixes, c'est-à-dire qu'on leur associe des descriptions discrètes. Par héritage, les niveaux supérieurs de la structuration héritent des images clés des plans.

Cette liaison permet tout à la fois d'ajouter le niveau des images dans une décomposition hiérarchique d'une vidéo et de gérer les détails sur les images comme une métadonnée parmi d'autres. En particulier, les contraintes d'associations entre éléments vidéo et métadonnées peuvent tenir compte de la classe *Image*. On peut ainsi contrôler que ce type de métadonnées n'est associé qu'aux feuilles des arbres de décomposition, mais rien n'empêche de généraliser le concept d'image(s) représentative(s) aux niveaux supérieurs, comme pour l'illustration des chapitres sur un DVD (*Digital Versatile Disc*).

5.2 Conclusion

Bien que nous nous intéressions à un domaine *a priori* limité, à savoir l'indexation des médias visuels du patrimoine culturel marocain, nous avons introduit dans ce chapitre un (méta)modèle d'indexation qui est assez général et flexible. En effet, il permet de décrire, dans une même collection, des documentaires, films, publicités, etc. Le modèle est assez générique et flexible caractérisé, d'une part, par une décomposition hiérarchique paramétrable

des types de vidéos et, d'autre part, par la variété des descripteurs que l'on peut associer à chaque décomposition d'une vidéo.

La liaison avec une base d'images a également été évoquée. Les images sont décrites par les mécanismes mis en œuvre par $Click^{Im}_{AGE}$.

Notre proposition n'est pas aussi générale que souhaitable pour certaines applications. Mais, soulignons qu'un modèle parfaitement général pour la vidéo n'existe pas ; *a priori*, et pour l'ensemble des propositions référencées, il faut disposer de toute la souplesse de modélisation offerte par des modèles de conception généralistes (relationnel, entités-associations, objets, etc.), notamment face à la diversité des métadonnées. Nous pensons que notre proposition offre un bon compromis entre modélisation et métamodélisation. Il peut se résumer à une séparation entre la structuration hiérarchique d'une vidéo, d'une part, et des métadonnées semi-structurées, d'autre part. La première partie est entièrement décrite dans la proposition. La seconde partie n'est introduite qu'indirectement *via* des contraintes d'associations entre nœuds de la hiérarchie et métadonnées élicitées.

Enfin, face à une importante quantité de documents visuels se pose la question de retrouver un document ou plusieurs documents répondant à une exigence particulière. L'interrogation formelle, *via* un langage de requêtes, est possible mais délicate avec les données temporelles [59, 32] et plus encore lorsque l'on combine des métadonnées sur le contenu (couleurs, textures mais aussi transitions visuelles pour la vidéo) avec des annotations (semi) structurées. La navigation est une technique issue des travaux sur les hypertextes qui a démontré, lorsqu'elle s'appuie sur une organisation pertinente des données, que la recherche dans une base est même plus aisée et efficace qu'en effectuant des requêtes [78].

Ainsi, le prochain chapitre présentera notre proposition sur cette forme d'interrogation dans une base de données de vidéos et d'images.

CHAPITRE 6

NAVIGATION CONJOINTE DANS UNE BASE D'IMAGES ET DE VIDÉOS

Au vu de la variété des approches et des modèles existant pour la recherche de documents visuels (cf. chapitre 3), on peut se demander laquelle adopter. À cette question, nous avons répondu par la recherche par la navigation. Notre choix est motivé par le fait que la recherche par la navigation convient bien à une recherche d'ordre général, un peu comme lorsque l'on parcourt une liste de nouvelles publications, une table des matières ou un index. On se déplace sur une base préalablement organisée.

Aussi, la navigation, comme la rétroaction, permet de combiner les apports du système avec les compétences visuelles de l'utilisateur. Les similitudes entre images (ou vidéos) sont stockées dans une base, et retrouver les images (ou vidéos) similaires à une image (ou vidéo) donnée est « instantané ». Le travail du système a été fait au préalable. Ensuite c'est l'œil de l'utilisateur qui permet de choisir une direction d'exploration plutôt qu'une autre. Si la technique de navigation elle-même peut être plus ou moins sophistiquée, le principal travail reste celui de la classification préalable.

Ce chapitre aborde ces différents aspects et présente notre proposition sur la navigation dans une base d'images et de vidéos. Nous y présentons une technique de navigation qui permet de naviguer en basculant indistinctement entre images et vidéos. Notre approche utilise les treillis de Galois implémenté dans $Click^{Im}_{AGE}$ pour la navigation dans une base d'images.

6.1 $Find^{Vi}_{DEO}$: un système générique pour la navigation dans une base de vidéos

Au chapitre 5 nous avons présenté un modèle assez générique et flexible caractérisé, d'une part, par une décomposition hiérarchique paramétrable des types de vidéos et, d'autre part, par la variété des descripteurs que l'on peut associer à chaque décomposition d'une vidéo. Pour aboutir à un système complet de recherche d'information visuelles, il convient d'ajouter une fonction d'interrogation à ce modèle.

Nous souhaitons élaborer en priorité un modèle de navigation se basant sur les treillis de Galois, comme cela a déjà été fait pour l'image seule [58] et que nous avons présenté au chapitre 4. Ce choix offre deux avantages intéressant. Tout d'abord, de par la structure complexe de la vidéo, partiellement connue et extensible pour les métadonnées, les langages adéquats font partie des langages d'interrogation des données semi-structurées, plus difficiles à manipuler que SQL. À l'opposé, la navigation peut être rendue intuitive et utilise les capacités de reconnaissance visuelle rapide de l'œil humain pour se déplacer efficacement sur les parties de la base connectées entre elles. Ensuite, les caractéristiques de bas niveau sont souvent comparées grâce à des mesures de similarité. Il est coûteux de parcourir l'ensemble d'une base importante de données de vidéos pour effectuer de tels calculs de similarité. À l'opposé, de nombreux calculs coûteux peuvent être effectués hors ligne et les résultats stockés dans le SGBD pour des temps de navigation optimaux (dans le meilleur des cas, cliquer sur un lien revient à suivre une clé étrangère).

Nous présentons dans cette section l'extension d'une instance du modèle de la figure 5.2 [66]. Le système résultant, que nous avons baptisé $Find^{Vi}_{DEO}$, est assez général. Il peut être facilement instancié en vue de définir un système de recheche dans une base de vidéos. Nous présentons dans la suite l'organisation des données et de la structure de navigation dans $Find^{Vi}_{DEO}$.

6.1.1 Organisation des données

Compte tenu du caractère temporel des vidéos ainsi que de la diversité des informations qui les composent, nous proposons une organisation de son contenu, puis nous nous basons sur cette organisation pour proposer une technique de navigation permettant dans un premier temps d'accéder de façon non-temporelle aux images de la vidéo et par la suite à la structure du document dans sa globalité.

Le modèle de la figure 5.2 peut être vu comme un modèle assez générique qui offre aux

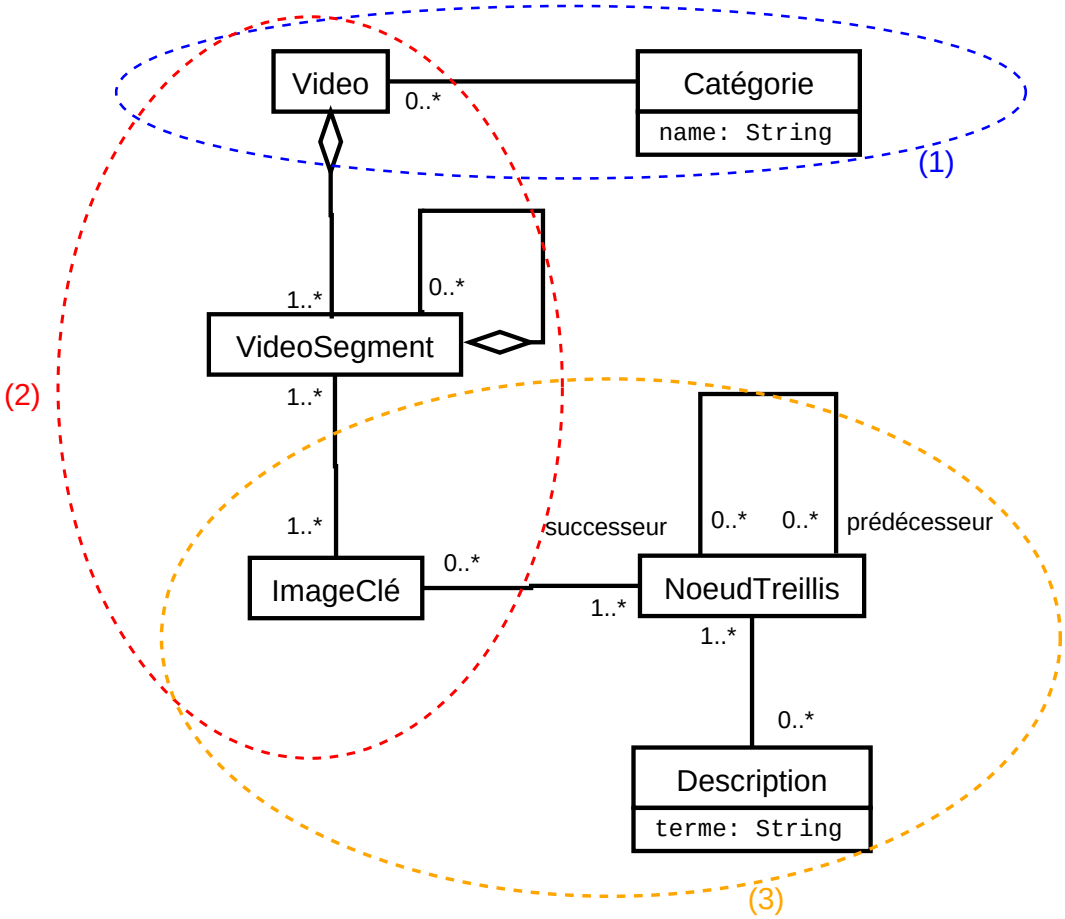

Figure 6.1 – Schéma UML de la base de vidéos

« indexeurs » la possibilité de définir, pour chaque type de vidéo, un schéma de description particulier. Ces schémas seront par la suite associés aux vidéos pour leur description.

D'autre part, alors qu'un schéma est déjà défini, il est possible de le faire évoluer en lui ajoutant de nouvelles entités. Ces dernières pourront ensuite être attachées à des vidéos pour leur description. La modification des schémas existants est cependant beaucoup plus complexe, du fait des nombreux impacts sur les entités la supportant.

En vue de la navigation, nous proposons (cf. figure 6.1) l'extension d'une instance du schéma de la figure 5.2. Il comprend quatre fonctions essentielles : (i) une classification des vidéos en catégorie, (ii) une structuration hiérarchique de profondeurs variables de chaque vidéo dont le niveau le plus bas est le plan, (iii) l'extraction d'une ou de plusieurs images clés par plan, (iv) une classification à l'aide de treillis de Galois des images clés issues des plans des vidéos.

6.1.2 Classification des vidéos

La classification des vidéos (cf. partie (1) de la figure 6.1) a pour objectif de mieux structurer la base de vidéos. La technique de classification que nous adoptons est une catégorisation manuelle des documents vidéos qui associe à chaque document une catégorie prédéfinie. Les documents vidéos associés à une même catégorie se trouvent regroupés ensemble. Pour créer une catégorie, il suffit de lui attribuer un nom. On peut ainsi avoir des catégories « film », « journal d'information », « documentaire », etc.

Ainsi, l'utilisateur peut choisir une catégorie, puis choisir les vidéos de cette catégorie, ou bien feuilleter l'ensemble des catégories. Une fois une catégorie déterminée, il peut effectuer la recherche d'un document particulier à l'intérieur de la catégorie.

6.1.3 Modélisation de la structure de navigation

L'objectif est de traduire le modèle de données vers un modèle de navigation. Il est important d'identifier le contenu des informations, les relations qui seront accessibles, mais aussi de distinguer les groupements possibles et les structures d'accès utilisables dans chaque cas.

Les modèles de conception d'hypermédia tels que RMM (*Relationship Management Methodology*) [41], OOHDM(*Object-Oriented Hypermedia Design Model*) [101], etc., sont inadaptés s'ils sont appliqués directement à notre modèle, ce qui impose une solution *ad hoc* que nous détaillons au chapitre 7, surtout pour les parties (2) et (3) de la figure 6.1.

Notre système propose les deux types de navigation tels que définis au chapitre 3 : la navigation intra vidéo et la navigation inter vidéo. La navigation intra vidéo consiste à naviguer sur une vidéo. Elle est définie sur les images clés extraites des plans. La navigation inter vidéos consiste à naviguer dans des bases de vidéos à travers leurs images clés.

6.1.3.1 Navigation inter vidéos

La technique de navigation inter vidéos (cf. partie (3) de la figure 6.1) que nous proposons permet de naviguer dans une base d'images clés extraites des plans des vidéos. Les images clés sont classifiées selon la technique du treillis de Galois. Le parcours du treillis permet de retrouver des images dont le contenu présente les caractéristiques visuelles requises. À partir d'une image, il est possible de retrouver le nœud du treillis qui la contient, une fois un nœud du treillis déterminé, le processus de navigation permet d'accéder aux autres nœuds du treillis. Ainsi, une fois une classe d'images délimitée, il suffit de retrouver le plan correspondant à l'une des images clés du cluster.

6.1.3.2 Navigation intra vidéo

La vidéo est composée d'un ensemble d'objets de base, organisés dans le temps et dans l'espace, sur lesquels différentes structures propices à la navigation peuvent être envisagées. Parmi ces structures, nous retenons les treillis de Galois, tels que définis précédemment, mais appliqués uniquement aux images clés extraites des plans d'une même vidéo. Cette méthode permet de naviguer de façon non-temporelle sur la vidéo. Une autre méthode est le parcours dans le temps de l'arborescence d'une vidéo avec une vue d'ensemble de celle-ci. Nous verrons dans la suite ces deux types de navigation.

Navigation intra vidéo temporelle : La méthode de navigation intra vidéo temporelle (cf. partie (2) de la figure 6.1) que nous avons retenue a été naturellement une méthode de base, à savoir le parcours de l'arborescence d'une vidéo avec une vue d'ensemble de celle-ci.

La structuration automatique, semi-automatique ou manuelle peut être utilisée. Pour illustrer une utilisation de notre modèle, nous avons adopté une segmentation semi-automatique des vidéos détaillée au chapitre 7. Les documents sont d'abord segmentés automatiquement en plan en utilisant une technique simple de comparaison d'histogrammes de couleur des images successives de la vidéo. Ensuite les plans sont regroupés manuellement pour former d'autres unités de granularité supérieure. Nous extrayons aussi automatiquement la première image de chaque plan pour être l'image clé du plan. Il s'agit donc d'une structure hiérarchique à deux niveaux, permettant d'accéder au contenu visuel d'une vidéo.

Une fois une vidéo trouvée, c'est-à-dire que l'utilisateur a validé sa requête en navigant sur la vidéo à l'aide de la technique des treillis de Galois décrite précédemment, il peut visualiser la vidéo à partir d'une image clé.

Navigation intra vidéo non-temporelle : Lors de la navigation intra vidéo temporelle, le parcours de la vidéo en vue de la validation de la requête d'un utilisateur est assez lent, car il faut attendre le chargement de la vidéo avant de la parcourir. Nous proposons de simplifier ce parcours en utilisant une approche *décentralisée* des treillis de Galois pour gérer les données. L'idée principale est de construire non plus un seul treillis pour l'ensembles des vidéos mais autant de treillis qu'il existe de vidéos en plus du treillis global. Chaque treillis créé est de taille réduite et les extensions des nœuds du treillis correspondront aux images d'une seule vidéo dans ce treillis. Le parcours de ce treillis permet une validation rapide des requêtes, car il offre la possibilité de naviguer de façon non-temporelle sur une vidéo.

Formellement, il s'agit de définir des sous-contextes relatifs à chaque vidéo. Si $\mathcal{V}$ repré-

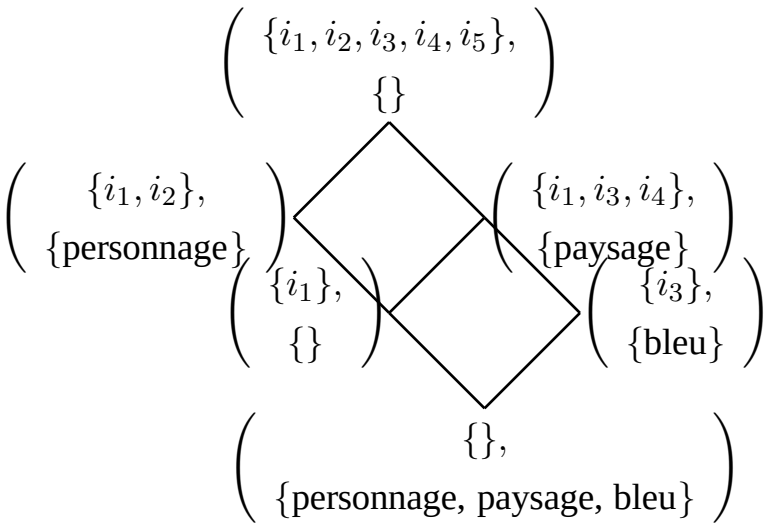

Figure 6.2 – Treillis d'héritage de la figure6.2

sente l'ensemble des vidéos alors $I = \{I_v\}_{v \in \mathcal{V}}$ est le sous-ensemble des images d'une vidéo v du contexte général $\mathcal{K}$. On définit ainsi la restriction de la relation $\mathcal{R}$ à la classe de vidéo v par $\mathcal{R}_v$. Ainsi, le contexte local $\mathcal{K}_v$ relatif à la vidéo v est le contexte $\mathcal{K}_v = (\mathcal{I}_v, \mathcal{D}, \mathcal{R}_v)$. Le treillis de Galois local est le treillis issu de ce contexte local.

La structure d'un tel treillis comporte plusieurs redondances (cf. figure 4.5). Or, notre objectif est de permettre une navigation rapide en vue de la validation de la requête de l'utilisateur. Il est donc préférable de représenter la même information de façon non-redondante et compacte. Pour cela, nous utilisons les treillis d'héritage selon les images clés.

Les treillis d'héritage tels que décrits dans Godin *et al.* [30] permettent d'éliminer dans un nœud les éléments dont l'extension apparaît dans tous ses ancêtres et l'intension dans tous ses descendants. Les éléments de l'extension ainsi éliminés sont alors retrouvés par héritage par rapport aux ancêtres, et les éléments de l'intension par héritage par rapport aux descendants.

L'héritage selon X est l'ensemble $r(x)$ tel que si on a $C = (X, X')$ alors :

$$r(X) = \{x \in E | x \in f(X') \wedge \nexists C' = (Y, Y') > C : x \in Y\}$$

Le treillis d'héritage selon X est donc l'ensemble des couples $(r(X), X')$ (cf. figure 6.2).

On remarque que les extensions des nœuds du treillis réduit peuvent être vides ou pas, comme le montre le troisième nœud de la partie gauche du schéma de la figure 6.2 (nœud (i_1, {}). Si le contenu réduit d'un nœud n'est pas vide, nous utilisons cette image pour représenter ce nœud. Si le contenu réduit d'un nœud est nul, la représentation du nœud est construite récursivement pour le contenu réduit de ses nœuds fils.

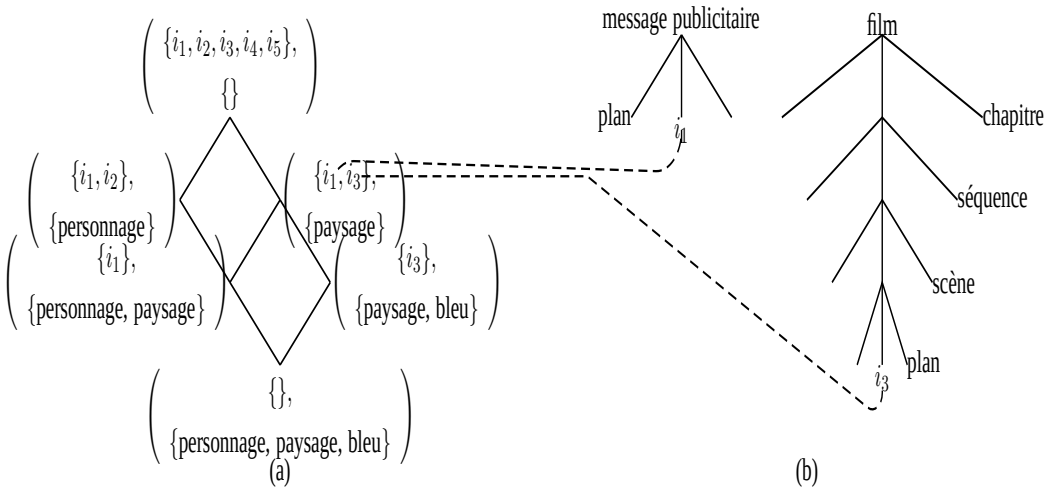

Figure 6.3 – Liaison entre vidéos et images clés du treillis

6.2 Navigation conjointe dans une base de vidéos et d'images

Nous avons présenté une technique de navigation dans une base d'images et une technique de navigation dans une base de vidéos. Or nous avons vu au chapitre 5 que la modélisation d'une vidéo peut être perçue comme une extension qui s'appuie sur la modélisation des images de manière quasi transparente. Nous avons aussi proposé une liaison entre la sous-base d'images et la sous-base des vidéos. Celle-ci est réalisée entre la classe *Image* du schéma de la sous-base d'image et la classe *ElémentVidéo* du schéma de la sous-base des vidéos à travers leurs images clés.

Nous proposons dans cette section, d'établir la liaison entre les deux méthodes de navigation ; le système $Find_{DIA}^{Me}$ résultant permet de naviguer en basculant indistinctement entre vidéos et images, et ce en s'appuyant aussi bien sur les ressemblances visuelles que sur la sémantique des images fixes ou animées.

Pour cela, nous appliquons la méthode de classification des treillis de Galois sur les images fixes et les images clés extraites des vidéos. Le schéma de la figure 6.3 illustre notre technique.

Le parcours du treillis permet de retrouver des images dont le contenu présente les caractéristiques visuelles requises. Une fois une classe d'images délimitée, si cette classe contient une image clé, il est possible de retrouver le plan correspondant à cette image clé. Inversement, à partir d'une image clé, on peut retrouver le nœud du treillis qui la contient et, indirectement, *via* un aller vers un cluster du treillis et un retour vers les vidéos, les autres

plans qui contiennent des images visuellement similaires.

Cependant, la construction du treillis de Galois est assez lente, la complexité étant en $O(n^2)$ où n est le nombre de nœuds. Mais, cet inconvénient n'a pas d'incidence sur les temps de réponse du système car la classification est effectuée hors ligne et les nœuds du treillis stockés. Cela permet de naviguer sur des données stockées, accélérant ainsi les temps de réponse en $O(1)$. Aussi, une fois le treillis créé, la complexité d'ajout de nouveau nœuds est seulement en $O(n)$ et celle d'ajout de nouvelles images à des nœuds du treillis existants seulement en $O(\log n)$.

6.3 Conclusion

Dans ce chapitre, nous avons proposé une approche générique pour la navigation dans une base de vidéos en exploitant leur contenu visuel (naviguer sur des bases de données structurées a déjà été largement étudié par ailleurs). Notre choix est motivé par le fait que, de par la structure complexe de la vidéo, partiellement connue et extensible pour les méta-données, les langages adéquats font partie des langages d'interrogation des données semi-structurées, plus difficiles à manipuler que SQL. Ensuite, les caractéristiques de bas niveau sont souvent comparées grâce à des mesures de similarité. Il est coûteux de parcourir l'ensemble d'une base importante de données de vidéos pour effectuer de tels calculs de similarité. À l'opposé, la navigation peut être rendue intuitive et utilise les capacités de reconnaissance visuelle rapide de l'œil humain pour se déplacer efficacement sur les parties de la base connectées entre elles. En plus, elle permet d'effectuer de nombreux calculs coûteux hors ligne et stocke les résultats dans le SGBD pour des temps de navigation optimaux.

La généricité de la structure permet de ne pas être dépendant d'une méthode particulière d'indexation. Notre proposition vise à naviguer en basculant entre unités des vidéos *via* leurs images clés et images fixes. Cette proposition, $Find^{Me}_{DIA}$, est issue de la liaison entre le (sous) $Find^{Vi}_{DEO}$ et le (sous) système $Click^{Im}_{AGE}$.

Avec $Find^{Vi}_{DEO}$, la navigation inter vidéos, permet de naviguer sur les images clés des vidéos situées sur différents nœuds du treillis. Il peut y avoir beaucoup de concepts et de correspondances entre les nœuds du treillis. Cette complexité du treillis est due à l'héritage multiple entre les concepts. Un nœud peut avoir plusieurs pères et plusieurs fils ce qui signifie qu'il peut être généralisé ou spécialisé de plusieurs manières différentes.

Nous avons également proposé des *storyboards* hiérarchisés qui ne sont rien d'autres que des résumés visuels des différentes parties d'une vidéo, permettant de naviguer séquentiellement sur une vidéo.

Nous validons la proposition dans un environnement de test au chapitre 7, comportant quelques dizaines de vidéos (générant des centaines d'images clés) et plusieurs centaines d'images fixes. Le *corpus* de documents visuels est spécialisé dans le patrimoine culturel marocain, notamment artisanal.

CHAPITRE 7

IMPLÉMENTATION ET EXPÉRIMENTATION

Comme nous l'avons vu au chapitre 2, les SRI traitent d'importantes quantités de données hétérogènes. Pour gérer ces informations, ses concepteurs ont besoin d'un modèle extensible et flexible. Les approches disponibles répondant, en partie, à ces besoins sont cependant très limitées dès qu'il s'agit de combiner différents types de média. Elles ne sont généralement pas construites sur une base de données, et quand c'est le cas, les systèmes de gestion de base de données relationnelles (SGBDR) sur lesquels elles sont construites ne disposent pas forcément de langage de requêtes pour ces médias. De plus, l'utilisation et l'optimisation des requêtes correspondantes ne sont pas évidentes.

Or, les possibilités offertes par les SGBDR telles que la différenciation entre schéma physique et schéma conceptuel et le langage de requêtes sont intéressantes pour les données visuelles. Mais, une problématique essentielle qui n'est pas des moindres, c'est que les données visuelles diffèrent sur de nombreux points des données usuellement utilisées par les SGBDR. Leur traitement exige la prise en compte de leur structure et de leur sémantique.

Étant donnés ces deux besoins contradictoires, nous avons choisi de combiner les avantages d'un modèle extensible et flexible avec une implémentation relationnelle. Nous bénéficions ainsi d'un modèle riche, mettant à disposition de nombreux types de haut niveau, et d'un moyen performant d'en interroger les données.

La section 7.1 présente l'architecture de $Find^{Me}_{DIA}$ d'un point de vue global dans un premier temps, ensuite nous décrirons les deux sous-systèmes qui le composent à savoir le sous-système $Find^{Vi}_{DEO}$ et le sous-système $Click^{Im}_{AGE}$. Enfin, la section 7.3 décrit le jeu de test que nous avons réalisé. L'objectif principal de ces expérimentations est d'évaluer l'intérêt de notre approche.

7.1 Architecture

Afin de tester la viabilité de l'ensemble de nos techniques nous avons réalisé le système $Find^{Me}_{DIA}$. Il est issue de la liaison entre deux sous-systèmes : Le sous-système $Find^{Vi}_{DEO}$ [66] permettant l'indexation des vidéos et le sous-système $Click^{Im}_{AGE}$ [64] permettant la navigation dans une base d'images. $Find^{Me}_{DIA}$ peut être vu comme une extension d'un SGBDR (PostgreSQL dans notre cas) au niveau de son langage d'interrogation. Il offre aux utilisateurs la possibilité :

1. de créer des types vidéos pouvant être attribués à plusieurs documents visuels de tailles variables;
2. de modéliser, d'indexer et de stocker ces données visuelles;
3. d'offrir aux utilisateurs la possibilité de jouer une vidéo, de visualiser une image et de lancer des requêtes d'interrogation sur les données stockées suivant des critères de recherche donnés;
4. de garantir des niveaux de qualité et de performance acceptables pour les utilisateurs.

7.1.1 Architecture globale de $Find^{Me}_{DIA}$

Nous présentons dans cette section une description globale des principes et des fonctionnalités du système sans rentrer dans la description de la structure interne qui sera présentée dans la section 7.1.2.

La figure 7.1 présente l'architecture globale de $Find^{Me}_{DIA}$. $Find^{Me}_{DIA}$ a été développé sous Linux, et au dessus du SGBDR PostgreSQL. Il utilise une interface écrite en PHP (Hypertext Preprocessor) et Java (Applet). Son architecture 3-tiers offre la possibilité de le déployer sur le Web permettant ainsi aux indexeurs et aux utilisateurs de manipuler les documents visuelles depuis des postes distants.

Le système comprend :

- une couche présentation présentant l'interface pour les indexeurs et les clients. Les indexeurs sont des spécialistes auxquels l'administrateur du système a donné le droit d'ajouter de nouveaux médias à la base et d'indexer ces médias. Les clients et les indexeurs sont des navigateurs Internet.
- une couche logique composée du serveur d'application qui communique avec les clients et les indexeurs par l'intermédiaire d'un serveur HTTP (Apache dans notre cas). C'est au niveau du serveur d'application qu'est implémenté les deux sous-

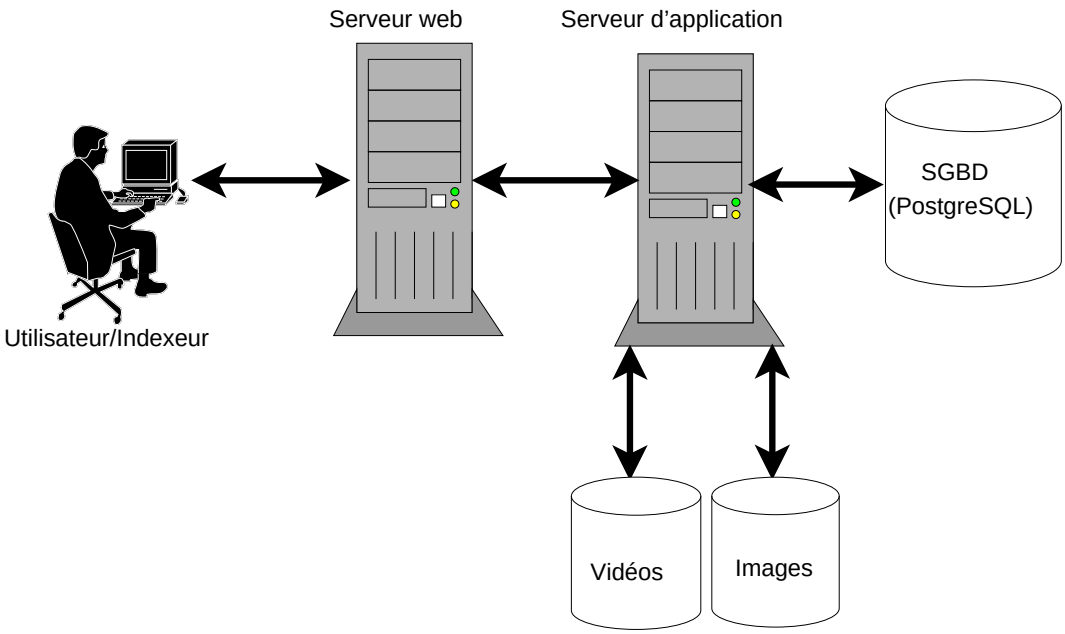

Figure 7.1 – Architecture globale de FindMedia

systèmes. Nous donnerons dans la suite une architecture détaillée de ces deux sous-systèmes.

- une couche physique chargée de la gestion des données comprenant une base de données (implémentée sous PostgreSQL), un système de fichiers composé des répertoires comportant les médias ainsi que des pages PHP générées lors de la création du treillis de Galois.

$Find^{Me}_{DIA}$ ne considère pas la base de données comme un seul support de stockage, mais au contraire étend le SGBD PostgreSQL afin de donner la possibilité aux indexeurs d'indexer l'image et la vidéo et aux utilisateurs la possibilité de rechercher ces média visuels.

Nous avons utilisé PostgreSQL comme SGBD, mais nous ne voulons pas être dépendant d'un SGBD particulier, c'est pourquoi nous avons choisi d'utiliser des ponts JDBC (Java DataBase Connectivity), qui nous permettent d'interroger et de peupler une base de données relationnelle quelconque. L'utilisation des ponts JDBC permet de voir une base de données PostgreSQL comme une base de données JDBC, ce qui permet d'être indépendant de ce logiciel et de le remplacer si besoin est par un autre logiciel.

Le schéma 7.2 présente la page d'accueil de $Find^{Me}_{DIA}$.

Cette page présente trois utilisateurs potentiels du système :l'administrateur, les des-

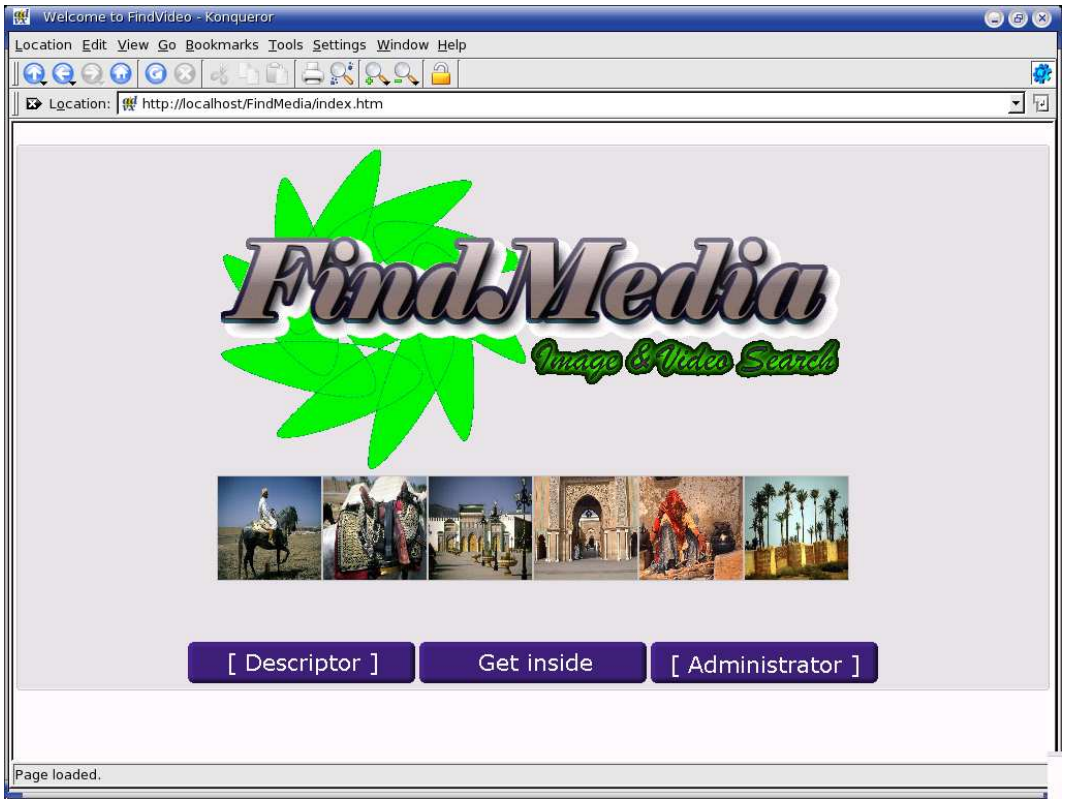

Figure 7.2 – Page d'accueil de $Find^{Me}_{DIA}$

cripteurs et les utilisateurs.

7.1.1.1 L'administrateur

L'administrateur est un super utilisateur qui a pour rôle la gestion de tout le système. Il gère les droits des autres utilisateurs en délivrant des autorisations.

7.1.1.2 Les descripteurs

Les descripteurs sont des utilisateurs ayant des droits particuliers. Ce sont des utilisateurs qui ont des connaissances en indexation de données multimédia. Après identification, ces utilisateurs peuvent grâce à une interface Web indexer des vidéos et des images depuis des postes distants.

7.1.1.3 Les utilisateurs

Les interfaces administrateur et descripteur ont deux vocations principales : d'une part, offrir à l'utilisateur les outils lui permettant de chercher des vidéos et des images par le biais

d'une navigation. D'autre part, lui permettre de visualiser les résultats correspondants à sa recherche. Les utilisateurs ont donc uniquement les droits de consultation, de visualisation et de téléchargement des vidéos et des images de la base.

7.1.2 Architecture détaillée de $Find^{Me}_{DIA}$

La figure 7.3 présente l'architecture détaillée de $Find^{Me}_{DIA}$. Cette architecture est issue de la liaison entre les deux sous-système $Find^{Vi}_{DEO}$ et $Click^{Im}_{AGE}$.

Les détails sur la liaison entre les deux sous-systèmes ont été expliqués au chapitre 5. Elle s'effectue entre la classe *Images*, provenant du schéma de la sous-base d'images, et la classe *ElémentsVidéos* du schéma de la sous-base de vidéos.

7.1.2.1 Le sous-système $Find^{Vi}_{DEO}$

$Find^{Vi}_{DEO}$offre les fonctionnalités pour la création de nouveaux types de vidéo, la gestion de grandes bases de données contenant plusieurs vidéos de taille variable, ainsi que des opérations de structuration, d'annotation et de recherche des vidéos. Nous détaillons dans la suite les différents modules pour l'indexation d'une vidéo.

Module de définition de types vidéo : La définition d'un type de vidéo, consiste à instancier le modèle de la figure 5.2 afin d'en dériver un sous-schéma pour ce type. Le type vidéo ainsi défini pourra être attribué aux vidéos pour leur description. En fait, Les types de vidéo ajoutés sont des instances de la relation *TypesVidéos*. La technique de définition de types que nous avons adopté est une catégorisation manuelle qui consiste à créer des types qui serviront à regrouper des vidéos de même type. Ainsi, les documents vidéos associés à un même type forme un même groupe. Pour créer un type, il suffit de lui attribuer un nom. On peut par exemple créer des types « dbfilm », « dbnews », « dbdoc », etc.

Module de structuration de types vidéo : Les types vidéo ajoutés sont structurés par la suite. La structuration consiste à associer aux types vidéo, des instances du *TypesElements-Vidéos*. Le nombre d'instance de *TypesElementsVidéos* associé à un type vidéo défini son niveau de structuration qui dépend des indexeurs.

Lors de la structuration du type vidéo, on peut associer des types de métadonnées à chaque niveau de la structuration et définir les champs pour ces métadonnées ce qui consiste à instancier la relation *MetaDonnées*.

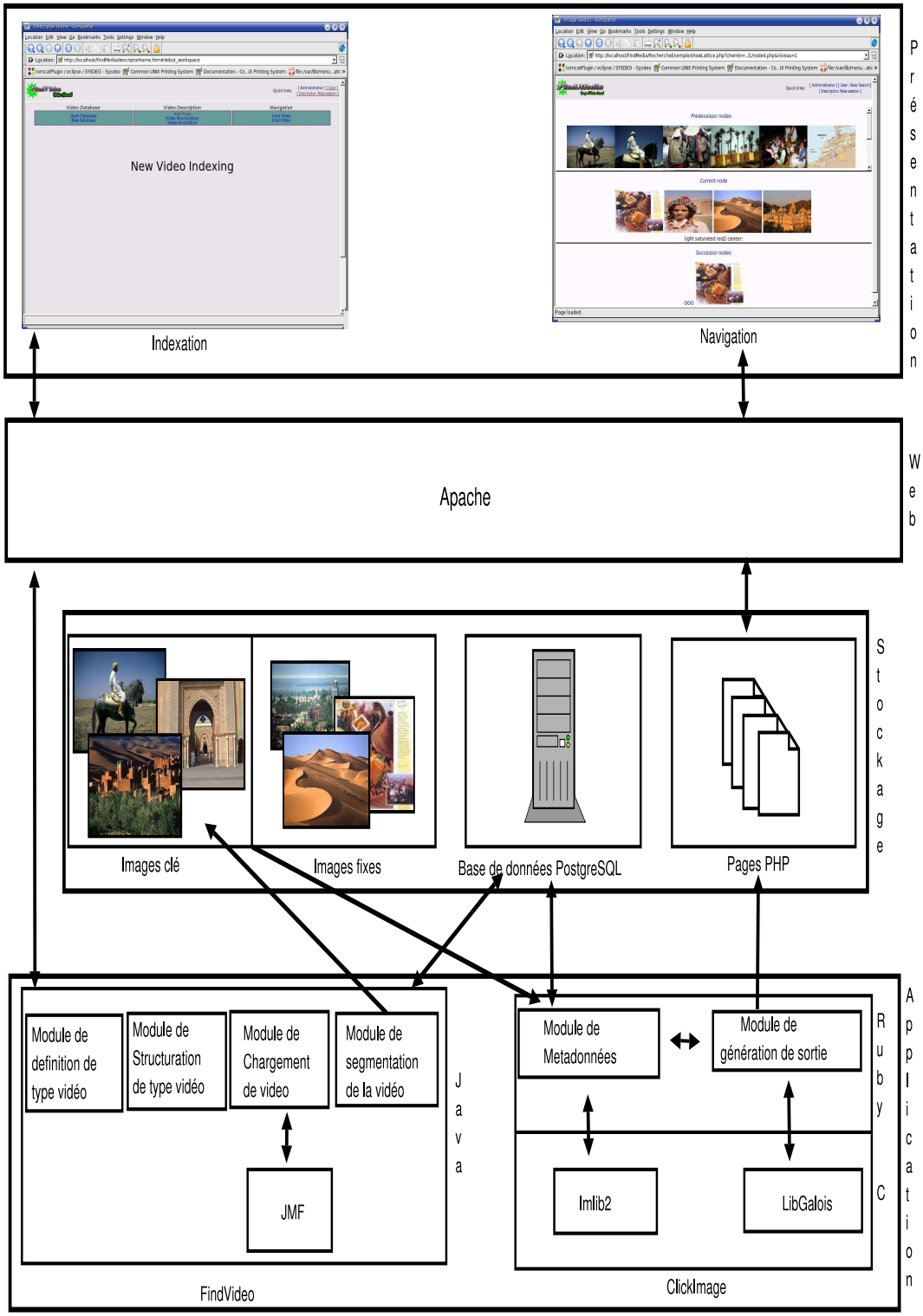

Figure 7.3 – Architecture détaillée de FindMedia

Module de Chargement des vidéos : Nous utilisons JMF (Java Media Framework) développe par Sun MicroSystems (dans sa version $2.1.1e$ [40]) pour le traitement et la structuration des vidéos. JMF est une architecture unifiée pour la synchronisation, le traitement, l'affichage de données temporelles comme les données vidéo, audio, le format MIDI etc. à l'intérieur d'applications indépendantes ou d'applets. Le choix de JMF est motivé par le fait qu'il permet de lire plusieurs formats de vidéos (AVI, MPEG, QT, H.261, H.263, Quick Time MOV) dans les applets comme dans les applications autonomes Java. Et aussi, il est modulaire et fournit la plupart des composants dont nous avons besoin. Enfin les spécifications de JMF indiquent les caractéristiques des classes à implémenter pour qu'elles puissent être traitées par java.

La spécification du JMF a commencé en 1996 et les premières implémentations ont été rendues public en 1997. Depuis la version 2.0, la capture, la sauvegarde, la transmission et le transcodage de l'audio et de la vidéo sont possibles.

La figure 7.4 montre une représentation des couches pour l'accès multimédia avec JMF et situe notre programme dans cette couche.

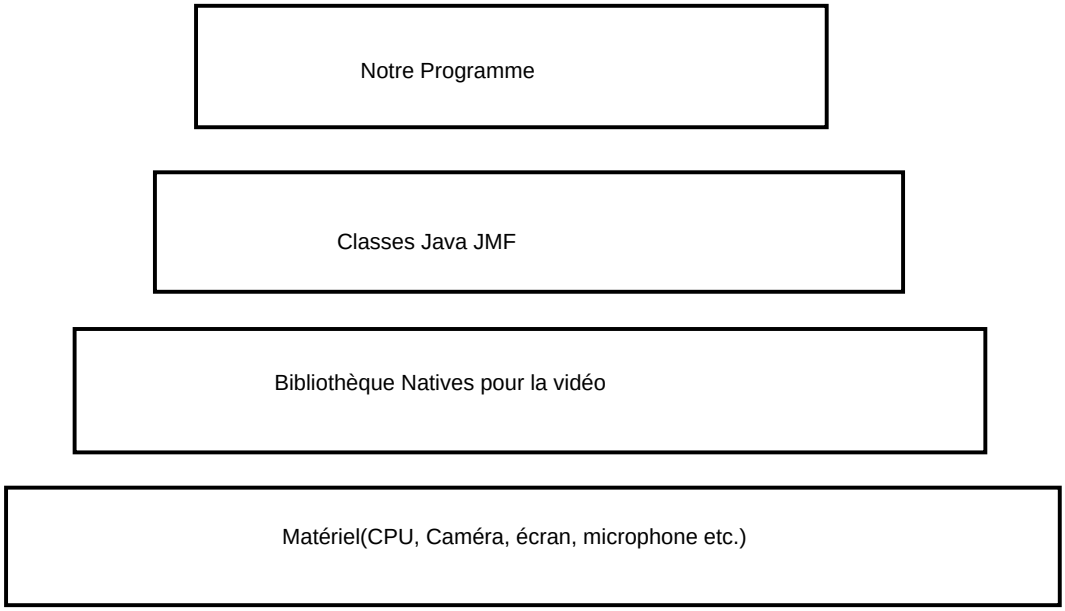

Figure 7.4 – Pile JMF pour la vidéo

JMF offre un accès aux fonctionnalités multimédia de haut niveau.

Module de structuration des vidéos : La figure 7.5 illustre la structure arborescente d'une vidéo avec une copie d'écran du module de structuration de $Find^{Vi}_{DEO}$. Il est possible d'attribuer à chaque vidéo, un type en sélectionnant un type de vidéo dans la liste déroulante

"Video Types". Il est ainsi possible de structurer cette vidéo conformement à la structure de ce type vidéo (la liste des niveaux, c'est à dire des vidéo éléments types). La structuration consiste en la segmentation en différents niveaux de granularité (plan, scène, ...) et l'extraction d'images clés.

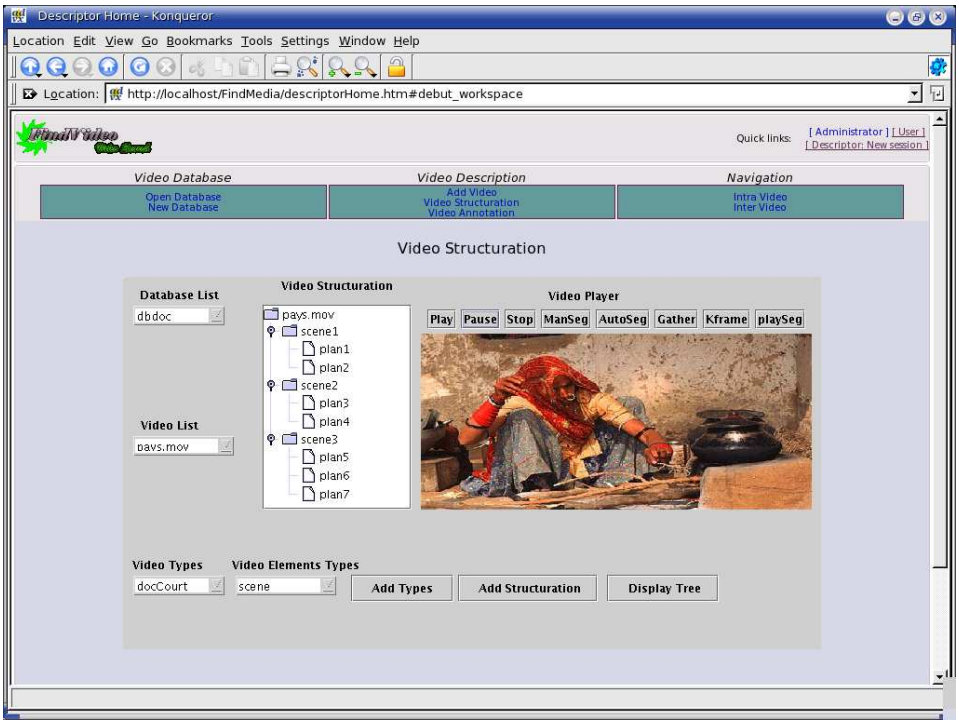

Figure 7.5 – Structuration et extraction d'images clés

L'indexeur a deux choix pour segmenter une vidéo en plan. Soit il procède manuellement en utilisant le bouton *ManSeg*, soit automatiquement en utilisant le bouton *AutoSeg*.

La segmentation automatique sera expliquée à la section 7.2.2. Une fois la vidéo segmentée en plan, le bouton *Gather* permet le regroupement manuel des plans en d'autres unités de granularités (scènes, séquences etc.). Par exemple les plans peuvent être regroupés en scènes, les scènes en séquences etc. Pour regrouper les plans en scène, il faudra afficher l'arbre de structuration et cliquer sur un plan qui sera le premier plan de la scène et un deuxième plan qui sera le dernier plan de la scène, et après cliquer sur le bouton *Gather*. Ainsi, les plans se trouvant entre les deux plans choisis constitueront les autres plans de la scène.

Le bouton *KFrame* permet d'extraire automatiquement la première image de chaque plan qui sera utilisée comme image clé de ce plan. Les niveaux supérieurs des plans (séquence, scène, ...) s'ils sont définis héritent des images clés de ses plans, afin d'offrir des

résumés visuels à l'utilisateur, similaires aux illustrations des chapitres sur un DVD.

Annotation des vidéos : Les annotations forment la base de l'indexation vidéo. Une annotation est une information associant de la sémantique à un ou à plusieurs segments d'une vidéo. Le schéma de la figure 7.6 montre quelques métadonnnées pouvant être associées à la racine de la vidéo.

Dans le cadre de cette thèse, l'annotation des vidéos est réalisée par $Click^{Im}_{AGE}$ à travers leurs images clés. Néanmoins, d'autres techniques d'annotation peuvent être ajoutées. Comme par exemple les résumés textuels qu'on peut ajouter à chaque niveau de la structuration de la vidéo. Dans le sens descendant, les annotations d'un niveau sont propagées vers tous ces subordonnées.

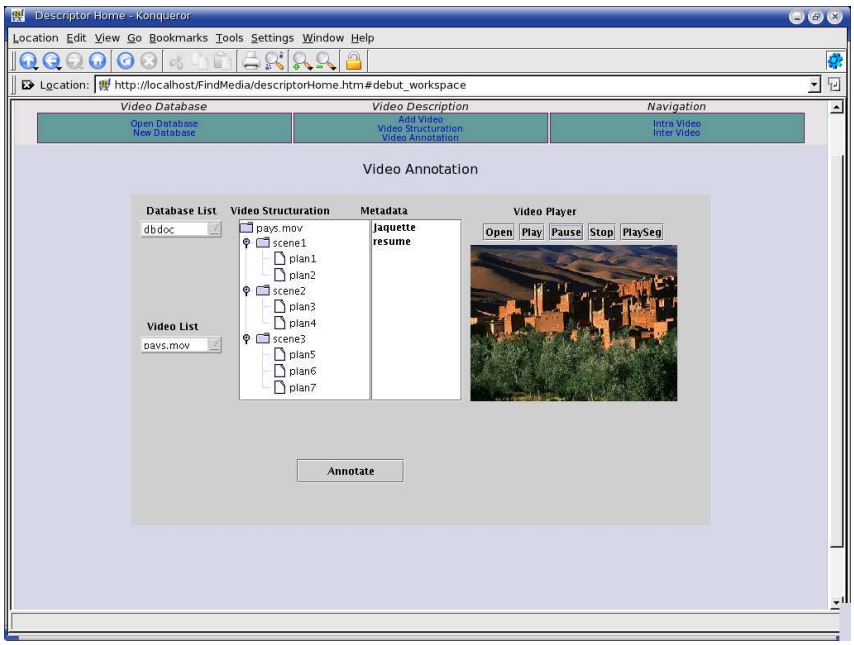

Figure 7.6 – Structuration hiérarchique d'une vidéo avec des métadonnées associées

7.1.2.2 Le sous-système $Click^{Im}_{AGE}$

Le sous-système $Click^{Im}_{AGE}$ est composé de trois modules essentiels : (1) LibGalois, une librairie écrite en C pour des raisons de performances, permettant de créer et de manipuler les treillis de Galois. Le treillis est créé incrémentalement en utilisant l'algorithme proposé par [30]. Les deux autres modules ont été écrits dans le langage de script Ruby: (2) le module de calcul de metadonnées qui utilise la librairie Imlib2 pour le traitement des

images et (3) le module de génération de sortie qui génère des pages PHP. Ce sont ces pages PHP qui sont utilisées pour la navigation dans la base de données.

7.2 Implémentation de la navigation

L'objectif est de fournir à l'utilisateur la représentation la plus claire possible de la navigation qu'il est susceptible de suivre pour aboutir à l'information recherchée.

La représentation hypermédia de notre technique s'inspire des méthodes spécialisées dans la conception d'hypermédias RMM (*Relationship Management Methodology*) [41], OOHDM (*Object-Oriented Hypermedia Design Model*) [101], etc. Ces méthodes proposent une conception par niveau où l'organisation des données et des parcours prennent une place prédominante. Elles proposent des étapes bien séparées, centrées sur trois aspects fondamentaux, à savoir : la modélisation des données de l'hypermédia, la procédure de découpage en tranches, et la modélisation de la structure de navigation c'est-à-dire l'adaptabilité aussi bien pour l'accès aux données que de leurs présentations. L'intérêt est de mieux structurer une application et de capturer la volatilité de l'information.

Généralement, le treillis résultant d'une relation d'indexation a une grande taille. Il n'est pas possible de visualiser directement toute la structure dans le processus de navigation. Nous adoptons l'approche qui consiste à montrer les prédécesseurs et successeurs immédiats par rapport au contexte courant de navigation.

Le point d'entrée pour le parcours du treillis est généralement l'un de ses sommets *« inf »* ou *« sup »*. Mais, compte tenu de la taille importante du nombre d'images du nœud *« sup »* (toutes les images) et du nombre de noeuds fils du nœud *« inf »*, nous proposons une méthode simple et rapide de recherche par l'exemple permettant de commencer la recherche à partir du premier nœud contenant une image requête. Ainsi, l'interrogation par l'exemple permet de retrouver des points d'entrée dans le treillis, puis la navigation permet d'explorer les nœuds du treillis.

D'un point de vue architectural, pour accéder à une vidéo représentée par une image clé, nous mettons un lien devant cette image permettant de visualiser la vidéo à partir de cette image.

Pour la visualisation du treillis, nous divisons l'écran d'affichage en trois parties. Le milieu affiche le nœud courant, le niveau supérieur affiche les nœuds précédents et le niveau inférieur les nœuds suivants.

La technique de recherche dans le treillis s'apparente à une technique de recherche par l'exemple combinée avec de la rétro-action. Les images du haut et du bas de l'écran sont

cliquables, c'est-à-dire que l'utilisateur pourra naviguer en cliquant sur une image qui ressemble le plus à celle qu'il cherche. Le système affiche alors le nœud courant correspondant et les nouveaux nœuds précédents et suivants. Les documents affichés subissent une évaluation de la part de l'utilisateur, s'il estime que l'image qu'il cherche n'est pas affichée, alors il peut cliquer sur une autre image plus proche de ses besoins. La boucle se poursuit jusqu'à ce que l'utilisateur soit satisfait ou qu'il estime que la base ne contient pas l'information recherchée.

La technique de navigation dans la base de données vidéo permet dans un premier temps d'accéder de façon non temporelle aux images de la vidéo grâce à la navigation inter vidéos et par la suite à la structure du document dans sa globalité grâce à la navigation intra vidéo.

7.2.1 Implémentation de la navigation Inter Vidéos

Notre technique de navigation inter vidéos utilise les images clés classifiées grâce à la technique des treillis de Galois décrite au chapitre 6. À partir d'une image clé, il est possible de retrouver le nœud du treillis qui la contient, une fois un nœud du treillis déterminé, le processus de navigation permet d'accéder aux autres nœuds du treillis.

La figure 7.7 montre le treillis de Galois avec une vue complète de toutes les images des nœuds d'un niveau du treillis. Les nœuds courants, précédents et suivants de ce niveau sont montrés à l'utilisateur.

7.2.2 Implémentation de la navigation Intra Vidéo

La première étape de la structuration d'un document vidéo en vue de la navigation intra vidéo consiste à découper la vidéo en unités temporelles de base que sont les plans. Comme nous l'avons souligné au chapitre 2, plusieurs méthodes de détection de plans existent. Certaines méthodes s'appuient sur la comparaison d'histogrammes de couleur [4, 55, 22] ou sur la comparaison pixel à pixel [51] des images successives, d'autres sont basées sur l'estimation du mouvement [14, 49]. Les avantages et les inconvénients de chacune de ces méthodes ont été discutés. Pour notre part, nous avons implémenté une méthode de structuration des documents vidéos basée sur les histogrammes de couleur. Notre choix est motivé d'une part par la rapidité des calculs d'une telle méthode et d'autre part par sa robustesse et sa précision. Notons tout de même que la principale difficulté de cette méthode est le choix du seuil qui permet de prendre des décisions pour segmenter la vidéo en un endroit précis. Au sein de notre laboratoire de l'ENSIAS de Rabat, [33] a développé une méthode robuste de segmentation des vidéos basée sur les histogrammes avec un seuillage utilisant l'entropie.

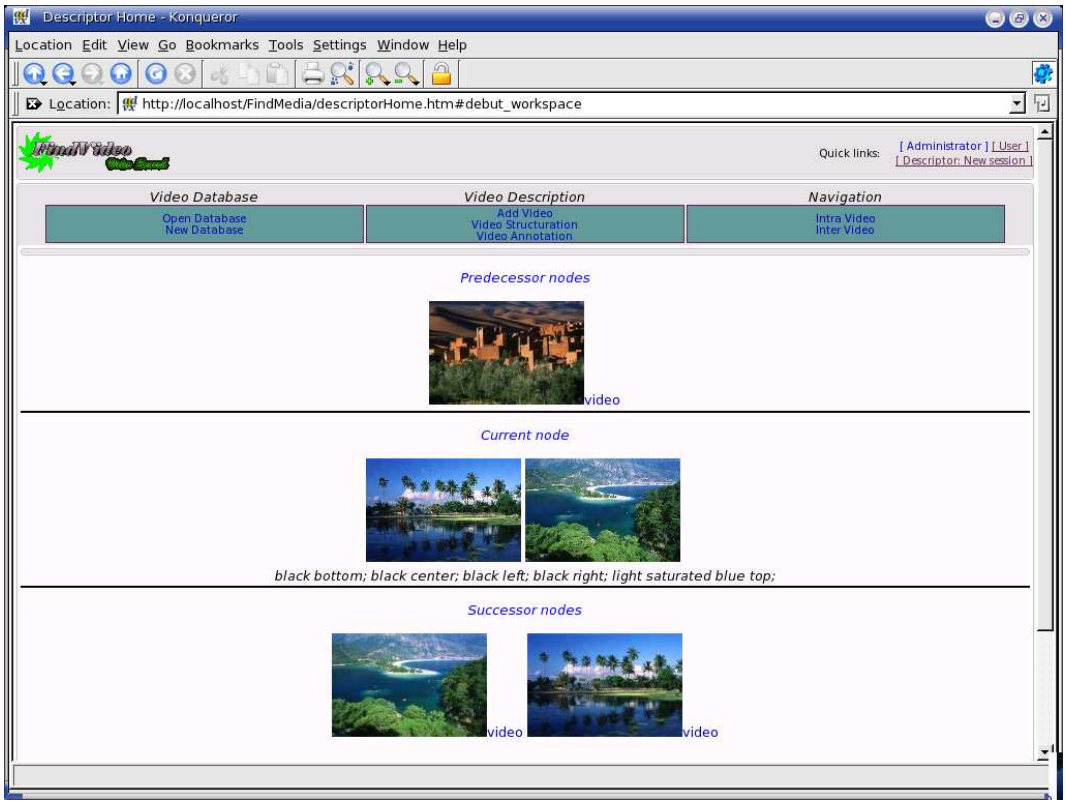

Figure 7.7 – Treillis de Galois pour la navigation dans une base de documents vidéos

Nous détaillons dans la suite cette méthode.

Rappelons d'abord que la méthode de détection par les histogrammes de couleurs mesure la distribution statistique des couleurs. Pour chaque couleur c, on compte le nombre de pixel de cette couleur par rapport au nombre de pixels total. Pour rendre un tel histogramme insensible et invariant aux différentes opérations géométriques telles que le changement d'échelle, en plus de la rotation et de la translation, les histogrammes sont normalisés.

La méthode de [33] utilise une transformation réversible des couleurs RVB. Cette transformation traite les trois composantes (R,V,B) en étudiant leur corrélation dans le domaine fréquentiel.

7.2.2.1 La transformation réversible de la couleur

Un codage typique des images consiste à attribuer à chaque pixel un triplet unique associé aux composantes rouge, verte et bleue (RVB). En général, chaque composante peut être codée sur 8 bits. Cette méthode associe chaque triplet (R, V, B) à une unique valeur entière $F(R, V, B)$. La particularité de cette transformation est qu'elle est bijective,

c'est-à-dire qu'elle permet de retrouver une couleur (R,V,B) à partir de $F(R,V,B)$.

Formellement, étant donné un entier n codé sur 8 bits, et m_i indiquant sa représentation à la i^{eme} bit. Un vecteur $U(n)$ est défini par :

$$U(n) = \sum_{i=1}^{8} m_i 2^{3(i-1)} \tag{7.1}$$

La transformation réversible de couleur F est définie comme suite :

$$F(R,V,B) = 4U(V) + 2U(R) + U(B) \tag{7.2}$$

Où les valeurs R, V, B sont codées sur 8 bits. On démontre facilement que la fonction mathématique F est une fonction bijective de $[0..255]^3$ vers $[0..2^{24}[$. Ainsi les valeurs prises par la transformée proposée peuvent être codées sur 24 bits.

Cette transformation est basée sur une décomposition binaire des composantes de couleur. L'idée principale est de trouver une transformation réversible. Cette décomposition est donnée à travers le vecteur U. Ainsi, la construction de $U(n)$ est basée sur la décomposition binaire du nombre n. Ce vecteur peut être calculé et stocké. Aussi, la transformation réversible de couleur F est rapide car c'est une fonction affine de $U(R), U(V), U(B))$.

On remarque que dans la formule de F, la composante verte est légèrement prioritaire par rapport aux composantes rouge et bleue. Ceci est due au fait que la perception de l'œil est plus sensible aux variations de la couleur verte qu'à celle du rouge ou du bleue.

7.2.2.2 Histogramme de la transformée de couleur réversible et entière (TRE)

Pour générer l'histogramme de couleur de la transformée de couleur réversible, le $F(R,V,B)$ de chaque pixel d'une frame est quantifié par M niveau. L'histogramme de couleur de la TRE $H_F(k)$ est donné par :

$$H_F(k) = \sum_{k=0}^{M} \delta(Q_F(F(R_{i,j}, V_{i,j}, B_{i,j})) - k), \tag{7.3}$$

$$\delta(i,j) = \begin{cases} 1, & \text{si } i = j \\ 0, & \text{si } i \neq j \end{cases} \tag{7.4}$$

Où, $Q_F()$ représente la fonction de quantification qui quantifie $F(R_{i,j}, G_{i,j}, B_{i,j})$ en une des M niveaux. La différence d'histogramme de couleur entre deux histogrammes est définie par :

$$dH_i = \sum_{j=0}^{M-1} \sum_{i=1}^{L-1} \left[\frac{(H_{i-1}(j) - H_i(j))}{H_{i-1}(j) + H_i(j)} \right]^2 \tag{7.5}$$

Où, L est le nombre total d'images dans la séquence vidéo, et dH la différence d'histogramme entre images successives.

7.2.2.3 Calcul de similarité entre images

Pour la détection de transition entre images d'une vidéo, on compare deux à deux les histogrammes de couleurs de ces images. Si la différence entre histogramme de couleur de deux images successives est supérieure à un seuil donné, alors on conclue qu'un changement de plan est présent à ce niveau de la séquence. Ainsi, la différence entre deux images se résume à une simple valeur, ce qui permet des calculs rapides. La détection des cuts et des transitions progressives, se ramène ainsi à l'étude de l'évolution de la valeur de cette valeur.

Il existe deux méthodes principales pour fixer le seuil. La première consiste à fixer un seuil après plusieurs expérimentations, la seconde méthode consiste à calculer automatiquement un seuil. Le calcul automatique du seuil est basé sur le contenu de la vidéo. Pour la segmentation temporelle de la vidéo, il est difficile de préfixer un seuil à cause des multitudes de styles de montage des vidéos d'une part et des multitudes de types de vidéos d'autre part. Un seuillage adaptatif est donc souhaitable. Par conséquent, [33] utilise une approche optimale appelée seuillage entropique qui permet de calculer un seuil optimal en utilisant la théorie de l'information.

7.2.2.4 Calcul du seuil en utilisant l'entropie

La méthode de seuillage entropique a été étendue pour pouvoir calculer l'entropie optimale pour la segmentation spatio-temporelle de la vidéo. Deux entropies résultent du calcul à partir de deux distributions de probabilité différentes. La première entropie qualifie les plans, la seconde les non-plan. Le seuil utilisé pour la segmentation est sélectionné de telle sorte que l'entropie totale soit maximisée. Le seuil est calculé comme suit :

$$soit f_k = \sum_{i=0}^{L-1} \sum_{k=0}^{M} \delta(dH_i - k) \tag{7.6}$$

$$P_{np}(i) = \sum_{i=0}^{T} \frac{f_i}{\sum_{k=0}^{T}} f_k \tag{7.7}$$

$$P_p(i) = \sum_{i=T+1}^{W} \frac{f_i}{\sum_{k=T+1}^{W}} f_k \tag{7.8}$$

$P_{np}(i)$ représente la probabilité pour les frames ne contenant pas une transition par rapport aux frames successives, et $P_p(i)$ la probabilité pour les frames contenant une transition par rapport aux frames successives. L'entropie correspondant aux deux classes est :

$$E_{np}(T) = -\sum_{i=0}^{T} P_{np}(i) \log P_{np}(i) \tag{7.9}$$

$$E_p(T) = -\sum_{j=T+1}^{T} P_p(j) \log P_p(j) \tag{7.10}$$

$E_{np}(T)$ représente les entropies pour ces deux classes séparées par un seuil T. Le seuil optimal T_{opt} est le seuil qui satisfait le critère suivant :

$$E(T_{opt}) = \sum_{T=0}^{W} max\{E_{np}(T) + E_p(T)\} \tag{7.11}$$

7.2.2.5 Extraction d'images clé

Une fois les plans segmentés, on procède à l'extraction d'une image clé, appelée aussi image représentative, pour chaque plan. Un plan étant une succession d'images sans interruption temporelle, il s'agit de choisir celle qui résume le contenu du plan. Notre but étant la classification des images clé extraites des plans par la méthode du treillis de Galois en utilisant les histogrammes de couleur, nous avons jugé suffisant de représenter chaque plan par uniquement une seule image, notamment la première image du plan qui peut être extraite manuellement ou automatiquement. Ce choix est lié au fait que généralement les images d'un même plan diffèrent peu par leur histogramme de couleur.

7.2.3 Implémentation de la navigation Intra Vidéo temporelle

La méthode de navigation intra vidéo temporelle que nous avons retenue a été naturellement une méthode de base, à savoir le parcours de l'arborescence d'une vidéo avec une vue d'ensemble de celle-ci.

Comme nous l'avons souligné, nous adoptons une segmentation automatique en plan et le regroupement manuel pour former d'autres unités de granularité. La profondeur de chaque arbre de segmentation est variable et dépend de la taille des vidéos ainsi que de leur type. Le plan est le niveau le plus bas de cette structuration. Ensuite nous extrayons une image clé par plan. Les segments vidéos ainsi extraits sont structurés pour former l'arbre de segmentation, et les images clés extraites des plans sont disposées sur un panel offrant la possibilité de naviguer sur la structure de la vidéo.

Il s'agit donc d'une structure hiérarchique à deux niveaux, permettant d'accéder au contenu visuel d'une vidéo. Une fois une vidéo choisie, cliquer sur un plan de l'arbre amène la vidéo à la première image du plan correspondant et affiche sur la barre « *keyframes* » les images clés correspondant à ce plan. Il est possible de lancer la vidéo à partir de ce point en cliquant sur le bouton « *Play* » ou seulement le plan correspondant en cliquant sur « *PlaySegment* », ou bien de se rendre directement à une image clé en cliquant sur cette image.

Si les plans sont regroupés pour former d'autres unités de granularité, cliquer sur toute unité amène la vidéo à la première image du premier plan correspondant de cette unité et affiche sur la barre « *keyframes* » les images clés des plans constituant cette unité.

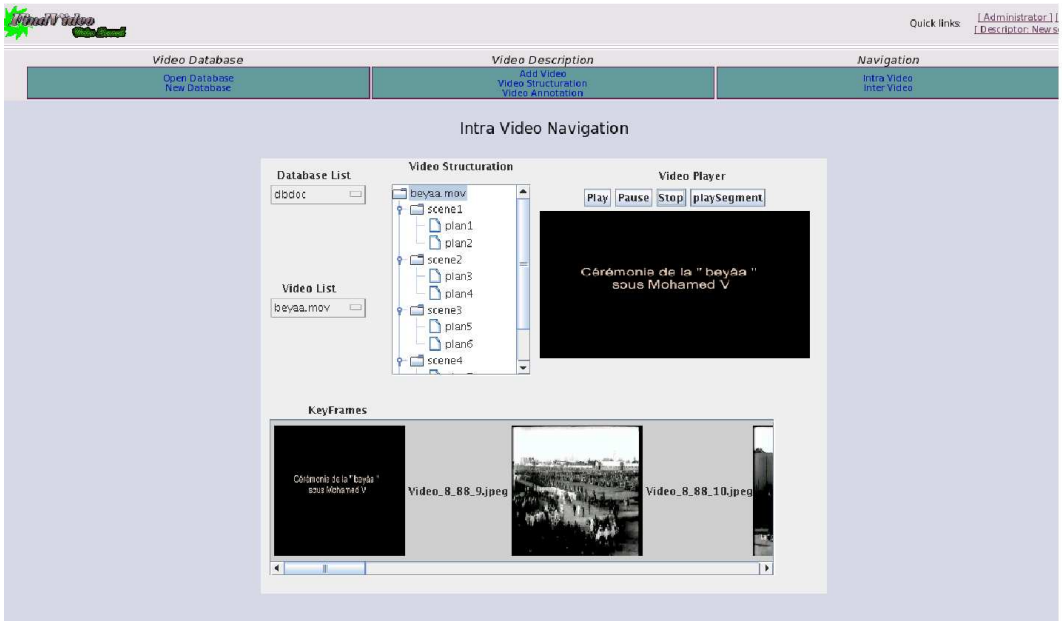

Figure 7.8 – Navigation intra vidéo

7.2.4 Implémentation de la navigation Intra Vidéo non-temporelle

L'implémentation de la navigation intra vidéo non-temporelle est semblable à celle de la navigation inter vidéos. La seule différence est que les images constituant le treillis de Galois dans le cas de la navigation intra vidéo sont des images clés des plans d'une même vidéo.

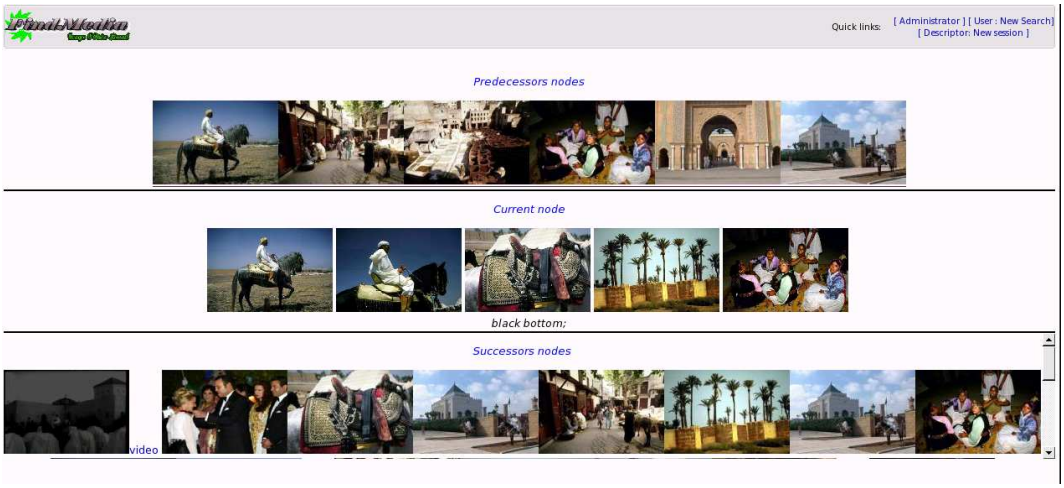

Figure 7.9 – Treillis de Galois pour la navigation dans une base de documents vidéos

7.2.5 Implémentation de la navigation conjointe

La figure 7.9 montre le treillis de Galois appliqué aux images fixes et aux images clés de la base. Les nœuds courants, précédents et suivants sont montrés à l'utilisateur. Il est possible de naviguer en basculant entre images fixes et segments vidéo à travers leurs images clés.

Le parcours du treillis permet de retrouver des images dont le contenu présente les caractéristiques visuelles requises. Une fois une classe d'images délimitée, on peut retrouver le plan correspondant à l'une des images clés du cluster. Inversement, à partir d'une image clé, on peut retrouver le nœud du treillis qui la contient et, indirectement, *via* un aller vers un cluster du treillis et un retour vers les vidéos, les autres plans qui contiennent des images visuellement similaires.

7.3 Expérimentation et résultats

Cette section décrit les expérimentations que nous avons faites afin d'évaluer l'intérêt de nos approches pour la navigation inter vidéos et pour la navigation conjointe. Nous avons donc effectué deux séries de test. Le test d'évaluation de la navigation inter vidéos a porté sur une base de données composée de vidéos et de leurs images clés. Au cours de cette évaluation, la navigation intra vidéo non-temporelle est aussi évaluée. Le test d'évaluation de la navigation conjointe a porté sur une base de données composée d'images fixes ainsi que de vidéos et de leurs images clés.

7.4 Évaluation de la navigation inter vidéos et intra vidéo non-temporelle

Le premier jeu de test comporte 15 vidéos du patrimoine culturel marocain. Les vidéos ont des durées comprises entre 10 et 15 minutes. Après avoir segmenté chaque vidéo en plan, nous avons extrait la première image de chaque plan. Formant ainsi une base de données composée de 150 images clés et 15 vidéos. Ensuite avec $Click^{Im}_{AGE}$, nous avons créé un treillis de Galois pour l'ensemble des images clés des vidéos. Pendant le processus de création du treillis de Galois global pour l'ensemble des images clés des vidéos, aussi, un treillis est construit pour chaque vidéo grâce à la méthode de l'approche décentralisée que nous avons présenté au chapitre 6. Ces treillis permettront la navigation intra vidéo non-temporelle.

Le tableau 7.1 montre le nombre moyen de propriétés par image indépendemment du treillis de Galois. Pour déterminer ce nombre moyen, α-cut a été fixé à 0.3. Ce choix permet de ne garder que les propriétés (la couleur dans notre cas) couvrant au moins 30% de la surface de la partie de l'image considérée. On remarque qu'avec α-cut fixé à 0.3, le nombre moyen de propriétés par image est compris entre 2 et 10, et que la plupart des images sont décrites par 5 propriétés. On peut donc conclure que quelqu'en soit le nombre d'images de la base, l'exploration du treillis en vue de la recherche d'une image ne depasse pas les 10 premiers nœuds.

Nombre de propriétés	1	2	3	4	5	6	7	8	9	10
Nombre d'images	1	1	6	8	56	27	19	23	6	3

Table 7.1 – Le nombre de propriétés par image

Le tableau 7.2 montre le nombre de niveaux et de nœuds du treillis de Galois global.

Nombre de vidéo	**Nombre image clé**	**Nombre de niveau**	**Nombre de nœud**
15	150	20	719

Table 7.2 – Nombre de niveaux et de nœuds pour les 15 vidéos de la base

Le tableau 7.3 détaille le nombre de nœuds présents à chaque niveau du treillis.

Le nombre total de propriétés retenu dans $Click^{Im}_{AGE}$ est de 34; par conséquent, le nœud inférieur du treillis se trouve au plus au 34^{eme} niveau (toutes les propriétés). Inversement le nœud supérieur se trouve au niveau 0 du treillis (aucune propriété). La disposition des autres nœuds dépend du nombre moyen de propriétés par image illustré sur le tableau 7.1.

niveau	nœud	niveau	nœud
0	1	7	64
1	20	8	24
2	78	9	8
3	115	10	2
4	157	11	1
5	148	12	1
6	113	20	1

Table 7.3 – Nombre de nœuds par niveaux dans le treillis de Galois

Le nombre moyen de propriété par image n'excédant pas 10, la plupart des nœuds sont donc situés dans les 10 premiers niveaux. Les niveaux centraux (niveau 3,4,5,6) contiennent beaucoup de nœuds. Il existe donc beaucoup d'interconnexions entre ces niveaux et leurs fils. Ce qui offre beaucoup de choix pour explorer une direction d'exploration plutôt qu'une autre. Accélérant ainsi les temps de recherche.

Pour l'évaluation de la navigation intra vidéo non-temporelle, le tableau 7.4 montre le nombre de niveaux et de nœuds pour chaque vidéo de la base.

Nous détaillons ici uniquement les résultats pour une vidéo (la vidéo 2). Les autres vidéos ont des résultats plus ou moins similaires. Le tableau 7.5 montre le nombre de nœuds présent à chaque niveau du treillis de la vidéo. Le nœud inférieur du treillis se trouve au $16^e me$ niveau (le nombre d'image n'est pas assez grand). Inversement le nœud supérieur se trouve au niveau 0 du treillis.

Le parcours de tel treillis en vue de la navigation intra vidéo non-temporelle est rapide car le nombre de niveaux et de nœuds de ce treillis ne sont pas élevés. On remarque que la plupart des nœuds sont situés dans les 9 premiers niveaux. Ainsi, l'exploration du treillis en vue de la recherche d'une image ne dépasse pas généralement les 9 premiers niveaux. Les niveaux centraux (niveau 3,4,5) contiennent beaucoup de nœuds. Il existe donc beaucoup d'interconnexions entre ces niveaux et leurs fils. Ce qui offre beaucoup de choix pour explorer une direction d'exploration plutôt qu'une autre.

7.5 Évaluation de la navigation conjointe

Le second jeu de test comporte les 150 images clés des 15 vidéos auxquelles nous avons ajouté 1050 images fixes. Nous avons donc une base composée de 15 vidéos, 150 images clés et 1050 images fixes. Ensuite avec $Click^{Im}_{AGE}$, nous avons créé un treillis de Galois

vidéo	image clé	niveau	nœud
1	19	14	41
2	22	16	59
3	9	7	9
4	7	9	25
5	7	8	19
6	7	8	19
7	11	9	56
8	10	9	24
9	10	5	8
10	9	8	32
11	6	8	17
12	18	14	102
13	4	7	15
14	3	2	2
15	8	8	12

Table 7.4 – Nombre de niveaux et de nœuds pour chaque vidéos de la base

pour ces 1200 images. Le tableau 7.6 donne les statistiques liées à cette évaluation.

Le nœud inférieur du treillis se trouve au plus au 34^{eme} niveau (toutes les propriétés). Inversement le nœud supérieur se trouve au niveau 0 du treillis (aucune propriété). La disposition des autres nœuds dépend du nombre moyen de propriétés par image.

Le tableau 7.7 montre le nombre de nœuds présent à chaque niveau du treillis.

On remarque que la plupart des nœuds sont situés dans les 10 premiers niveaux. Ainsi, l'exploration du treillis en vue de la recherche d'une image ne dépasse pas généralement ces 10 premiers niveaux. Les niveaux centraux (niveau 3,4,5,6,7) contiennent beaucoup de nœuds comparés au nombre total d'images indexées. Il existe donc beaucoup d'interconnexions entre ces niveaux et leur fils. Ce qui offre beaucoup de choix pour explorer une direction d'exploration plutôt qu'une autre. Accélérant ainsi les temps de recherche.

7.6 Conclusion

Nous avons introduit dans ce chapitre une implémentation de $Find^{Me}_{DIA}$. Nous avons choisi de combiner les avantages d'un modèle extensible et flexible avec une implémenta-

niveau	nœud	niveau	nœud
0	1	7	5
1	6	8	3
2	6	9	3
3	9	10	1
4	8	11	1
5	8	12	1
6	3	16	1

Table 7.5 – Nombre de niveaux et de nœuds dans le treillis de Galois pour la vidéo 2

vidéo	image clé	image fixe	niveau	nøeud
15	150	1050	34	5198

Table 7.6 – Treillis de galois pour 150 images clés et 1050 images fixes

niveau	nøeud	niveau	nøeud
0	1	7	756
1	44	8	305
2	304	9	118
3	966	10	24
4	1412	11	4
5	1365	12	4
6	1110	34	1

Table 7.7 – Nombre de niveaux et de nœuds dans le treillis de Galois

tion relationnelle. Nous bénéficions ainsi d'un modèle riche, mettant à disposition de nombreux types de haut niveau, et d'un moyen performant d'en interroger les données.

Nous avons validé nos approches à travers des expérimentations sur des ensembles d'images et de vidéos du patrimoine culturel marocain. Ces tests de performances effectués montrent que la demande initiale de tenue en charge est respectée.

CHAPITRE 8

CONCLUSION GÉNÉRALE

Les systèmes de recherche d'informations visuelles visent la maîtrise d'une nouvelle frontière dans l'univers d'informations numériques, celles d'images fixes et animées. Il s'agit d'un défi majestueux et prometteur car il nécessite des efforts interdisciplinaires, en bases de données, en analyse et traitement d'images et en recherche d'informations documentaires, etc. pour la réalisation d'une telle édifice.

Ce chapitre résume nos principales contributions à cette construction. Mais expose aussi les limites de nos approches employées.

8.1 Contributions principales

Comme nous l'avons introduit au début de ce mémoire, cette thèse a été réalisée dans le but de répondre à un problème concret à savoir la conservation et surtout l'indexation du patrimoine culturel marocain filmé et photographié.

Nous avons tout d'abord abordé le problème de l'indexation multimédia. Les principales contributions de nos travaux concernent la définition d'outils génériques pour la mise en œuvre de systèmes spécifiques d'indexation.

Nous avons d'abord introduit un (méta)modèle d'indexation des vidéos qui est assez général et flexible. Ensuite nous avons présenté une méthode de navigation dans une base d'images et de vidéos qui tente d'effectuer une synthèse difficile des propositions existantes. Nous présentons dans la suite les apports de nos méthodes vis-à -vis de ces deux aspects.

8.1.1 Apport vis-à -vis de l'organisation des données

L'approche que nous avons adoptée, utilise les techniques rencontrées dans la littérature sur les bases de vidéos mais aussi dans d'autres domaines.

Nous avons d'abord introduit un (méta)modèle d'indexation des vidéos qui est assez général et flexible. En effet, il permet de décrire, dans une même collection, des documentaires, films, publicités, etc. De la même façon, à chaque élément de la vidéo, il permet d'associer les méta-données nécessaires à sa description, en évitant que toute métadonnée puisse indexer tout élément de la vidéo.

L'apport principal de notre méthode à ce niveau se traduit, d'une part, par sa flexibilité, c'est-à -dire par la capacité du système à s'adapter à différents types de vidéos, par une décomposition hiérarchique paramétrable des vidéos et, d'autre part, par la variété des descripteurs que l'on peut associer à chaque image clé extraite des plans. De plus, la classification des images clés par la technique des treillis de Galois est effectuée hors ligne ce qui permet de naviguer sur des nœuds précalculés et structurés, accélérant ainsi les temps de réponse. L'algorithme retenu étant incrémental, le treillis n'est toutefois pas figé, ce qui permet d'insérer au fur et à mesure les images clés de nouvelles vidéos.

8.1.2 Apport vis-à -vis de la navigation

Face à une importante quantité de documents visuels se pose la question de retrouver un document ou plusieurs documents répondant à une exigence particulière. Des interfaces d'interrogation graphique [26, 7] ou des langages de requêtes [32, 36] (exploitant les relations temporelles d'Allen [5]), voire seulement SQL [61], ont déjà été proposés par ailleurs. Nos travaux sont plutôt orientés vers le grand public, par conséquent, c'est la navigation que nous avons retenue comme moyen de recherche dans la base de données visuelles. En effet, la navigation est une technique issue des travaux sur les hypertextes qui a démontré, lorsqu'elle s'appuie sur une organisation pertinente des données, que la recherche dans une base est même plus aisée et efficace qu'en effectuant des requêtes [78].

Les techniques de navigation rencontrées dans la littérature sont généralement basées sur une structuration des éléments vidéos sous forme d'arbre ou de graphe de voisinage. L'inconvénient de telles méthodes est l'écueil de l'arborescence où chaque « erreur » dans le choix d'un nœud fils nécessite des retours en arrière pour descendre dans une nouvelle branche. Dans le cas des graphes de similarité deux à deux, cet écueil n'existe pas mais la navigation se fait « point à point » c'est-à -dire qu'à partir d'une image de référence, on navigue uniquement vers une autre image voisine au lieu d'accéder à une sous-classe d'images présentant des similitudes.

La technique de classification par un treillis de Galois que nous avons proposé offre de nombreux avantages. De par ses nombreuses interconnexions, elle évite l'écueil de l'arborescence. Elle permet la navigation ensembliste, qui permet à partir d'un nœud du treillis

constitué d'un ensemble d'images de naviguer vers d'autres nœuds voisins présentant des similitudes. Elle permet des recherches conjonctives sur l'ensemble des métadonnées élicitées, en ne privilégiant aucun point de vue. Aussi, les recherches sont dichotomiques en le nombre de propriétés retenues.

Ainsi, la méthode de navigattion que nous avons proposé peut être vue comme un modèle assez générique supportant les deux principales techniques de navigation à savoir la navigation intra vidéo et la navigation inter vidéos, toutes deux basées sur une modélisation des données et de la structure de navigation.

Cependant, la construction du treillis de Galois est assez lente, la complexité étant en $O(n^2)$ où n est le nombre de nœud. Mais, cet inconvénient n'a pas d'incidence sur les temps de réponse du système car la classification est effectuée hors ligne et les nœuds du treillis stockés. Cela permet de naviguer sur des données stockées, accélérant ainsi les temps de réponse en $O(1)$. Aussi, une fois le treillis créé, la complexité d'ajout de nouveau nœuds est seulement en $O(n)$ et celle d'ajout de nouvelles images aux nœuds du treillis seulement en $O(\log n)$.

Nous avons finalement validé nos approches en implémentant ces différentes méthodes au sein du prototype $Find^{Me}_{DIA}$. $Find^{Me}_{DIA}$ a été dévéloppé sous Linux, et au dessus du système de Gestion de Bases de Données Relationnel(SGBDR) PostgreSQL. En effet, en plus d'être matures et très répandus, les SGBDR présentent des performances et des possibilités de montée en charge en adéquation avec les besoins exprimés.

8.2 Limites de l'approche

Bien qu'il réponde correctement aux besoins exprimés, notre modèle souffre de certaines limitations. En effet, Notre modèle n'est pas aussi général que souhaitable pour certaines applications. Mais, soulignons qu'un modèle parfaitement général pour la vidéo n'existe pas ; *a priori*, et pour l'ensemble des propositions référencées, il faut disposer de toute la souplesse de modélisation offerte par des modèles de conception généralistes (relationnel, entités-associations, objets, etc.), notamment face à la diversité des métadonnées. Nous pensons que notre modèle offre un bon compromis entre modélisation et métamodélisation. Il peut se résumer à une séparation entre la structuration hiérarchique d'une vidéo, d'une part, et des métadonnées semi-structurées, d'autre part. La première partie est entièrement décrite au chapitre 2. La seconde partie n'est introduite qu'indirectement *via* des contraintes d'associations entre nœuds de la hiérarchie et méta-données élicitées.

L'implémentation au dessus d'un SGBD relationnel offre de nombreux avantages. Ce-

pendant les SGBD relationnels sont limités dès qu'il s'agit de representer des média visuels.

Face à ces difficultés, nous avons choisi de combiner les avantages d'un modèle extensible et flexible avec une implémentation relationnelle. Cette approche hybride employée demande un investissement important. En plus d'une bonne maîtrise des modèles relationnels, elle nécessite en effet l'apprentissage de nouveaux concepts.

8.3 Perspectives

Les persepectives que nous envisageons dans le prolongement de ces travaux de thèse s'articule autour de l'automatisation des outils d'indexation et l'ajout automatique ou manuel d'annotations sémantiques. Les annotations sémantiques rendront les métadonnées plus ou moins riches. La notion de strates, classifiées, extensibles, multi-attributs et accompagnées d'informations spatio-temporelles nous paraît une base assez générale. L'extension serait une véritable ontologie. La prise en compte de ces annotations permet d'enrichir la requête des utilisateurs dans le but d'avoir des résultats pertinents tout en respectant l'intérêt des utilisateurs. Il convient donc d'ajouter un module d'interrogation par le contenu. L'interrogation formelle, *via* un langage de requêtes, est possible mais délicate avec les données temporelles [59, 32] et plus encore lorsque l'on combine des métadonnées sur le contenu (couleurs, textures mais aussi transitions visuelles pour la vidéo) avec des annotations (semi) structurées.

Bibliographie

[1] S. ABITEBOUL. Object databases support for digital libraries. In SPRINGER-VERLAG, réd., *Proceedings of the 1st European Conference on Research and Advanced Technology for Digital Libraries (ECDL'97)*, number 1324 in Lecture Notes in Computer Science (LNCS), pages 39–45, Pisa, Italy, 1–3 septembre 1997.

[2] B. ADAMS, C. DORAI et S. VENKATESH. Novel approach to determining tempo and dramatic story sections in motion pictures. In *IEEE International Conference on Image Processing ICIP*, Vancouver, Canada, 2000.

[3] G. AHANGER et T. D. C. LITTLE. A system for customized news delivery from video archives. In 4^{th} *International Conference on Multimedia Computing and Systems (ICMCS)*, pages 526–533, Ottawa, Canada, 1997.

[4] M. AHMED, S. ABU-HAKIMA et A. KARMOUCH. Key frame extraction and indexing for multimedia databases. In *Visual Interface 99 Conference*, pages 506–511, May 1999.

[5] J. F. ALLEN. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–543, septembre 1983.

[6] H. AOKI et O. HORI. A shot classification method of selecting effective key-frames for video browsing. In *ACM Multimedia*, pages 1–10, Boston, Massachusetts, 1996.

[7] E. ARDIZZONE, L. C. MARCO, U. MANISCALCO, D. PERI et R. PIRRONE. Population and query interface for a content-based video database. In *International Workshop on Multimedia Data and Document Engineering (MDDE)*, Prague, Czech Republic, mars 2002.

[8] F. ARMAN, R. DEPOMMIER, A. HSU et M.-Y. CHIU. Content-based browsing of video sequences. In *Proceedings of the second ACM international conference on Multimedia*, pages 97–103, New York, NY, USA, 1994.

[9] D. H. BALLARD et C. M. BROWN. *Computer Vision*. Prentice-Hall, 1982. 523 p.,

[10] M. BARBUT et B. MONJARDET. *Ordre et classification — Algèbre et combinatoire (2 tomes)*. Hachette, 1970.

[11] E. BERTINO, B. C. OOI, R. SACKS-DAVIS, K. L. TAN, J. ZOBEL, B. SHIDLOVSKY et B. CATANIA. *Indexing Techniques for Advanced Database Systems*. Kluwer Academic Publishers, Boston, 1997. 250 pages,

[12] M. BOUET, A. KENCHAF et H. BRIAND. Shape representation for image retrieval. In *Proceedings of the 6th ACM International Multimedia Conference (ACM-MM'99)*, volume 2, pages 1–4, Orlando, Floride, novembre 1999.

[13] P. BOUTHEMY, Y. DUFOURNAUD, R. FABLET, R. MOHR, S. PELEG et A. ZOMET. Video hyper-link creation for content-based browsing and navigation. In *Workshop on Content-Based Multimedia Indexing, CBMI'99*, Toulouse, France, octobre 1999.

[14] J. CALIC, S. SAV et E. IZQUIERDO. Temporal video segmentation for real-time key frame extraction. In *ICASSP*, 2002.

[15] C. CARSON et V. E. OGLE. Storage and retrieval of feature data for a very large online image collection. In *IEEE Computer Society Bulletin of the Technical Committee on Data Engineering*, volume 19, pages 19–27, décembre 1996.

[16] G.-H. CHA et C.-W. CHUNG. Object-oriented retrieval mechanism for semistructured image collection. In *Proceedings of the 6th ACM International Multimedia Conference (ACM-MM'98)*, Bristol, Royaumes-Unis, septembre 1998.

[17] Y. CHAHIR. *Indexation et recherche par le contenu d'informations visuelles*. Thèse de Doctorat, ICTT, Lyon, France, 1999.

[18] S. F. CHANG, J. R. SMITH, M. BEIGI et A. BENITEZ. Visual information retrieval from large distributed online repositories. *Communications of the ACM*, 40(12):63–67, 1997.

[19] C. CHRISMENT et F. SÉDES. Toward a unified view of media annotation. *Ingénierie des systèmes d'informations*, 7(5–6):45–63, 2002.

[20] M. CHRISTEL et A. WARMACK. The effect of text in storyboards for video navigation. In *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, Salt Lake City, USA, mai 2001.

[21] M. CORRIDONI, A. Del BIMBO et P. PALA. Image retrieval by color semantics. *Multimedia Systems*, 7(3):175–183, 1999. Springer-Verlag,

[22] A. DAILIANAS, R. B. ALLEN et P. ENGLAND. Comparaison of automatic video segmentation algorithms. In *Photonics West*, pages 2–16, Philadelphia, Pennsylvania, octobre 1995.

[23] G. DAVENPORT, T.A. SMITH et N. PINCEVER. Cinematic primitives for multimedia. In *IEEE Computer Graphics & Applications*, pages 67–74, 2002.

[24] C. FALOUTSOS, R. BARBER, M. FLICKNER, J. HAFNER, W. NIBLACK et D. PETKOVIC. Efficient and effective querying by image content. *Journal of Intelligent Information Systems*, 3(3-4):231–262, 1994.

[25] S. FISCHER, R. LIENHART et W. EFFELSBERG. Automatic recognition of film genres. In *Proceedings ACM Multimedia retrieval*, 1995.

[26] M. FLICKNER, H. SAWHNERY, W. NIBLACK, J. ASHLEY, Q. HUANG, B. DOM, M. GORKANI, J. HAFNER, D. LEE, D. PETKOVIC, D. STEELE et P.YANKER. Query by image and video content: The QBIC system. *IEEE Computer*, 28(9):23–32, septembre 1995.

[27] W. B. FRAKES et R. BÄEZA-YATES, réds. *Information Retrieval: Data Structures & Algorithms*. Prentice-Hall, 1992. 504 p.,

[28] T. GEVERS. *Colour Image Invariant Segmentation and Retrival*. Thèse de Doctorat, University of Amsterdam, Netherlands, mai 1996. 142 pages,

[29] R. GODIN, G. MINEAU et R. MISSAOUI. Méthodes de classification conceptuelle basées sur les treillis de galois. *Revue dintelligence artificielle*, 9(2):105–137, 1996.

[30] R. GODIN, R. MISSAOUI et H. ALAOUI. Incremental concept formation algorithms based on galois (concept) lattices. *Computational Intelligence*, 11(2):246–267, 1995.

[31] A.S. GORDON et E.A DOMESHEK. Retrieval interfaces for video databases. In *R. Burke. (ed.), Proceedings of the AAAI fall Symposium on AI applications in knowledge Navigation and retrieval, Massachusetts Institue of Technology*, pages 45–51, 10–12 novembre 1995.

[32] M. HACID, C. DECLEIR et J. KOULOUMDJIAN. A database approach for modeling and querying video data. *IEEE Transactions on Knowledge and Data Engineering*, 12(5):729–750, septembre 2000.

[33] Y. HADI, F. ESSANNOUNI, R.O.H THAMI, A. SALAM et D. ABOUTAJDINE. A new approach for video cut detection using color histogram. In *International Symposium on Image/Video Communications, ISIVC'2006*, 13-15 septembre 2006.

[34] D. HEESCH, P. HOWARTH, J. MAGALHÀES, A. MAY, M. PICKERING, A. YAVLINSKY et S. RÜGER. Video retrieval using search and browsing. In *TREC Video Retrieval Evaluation Online Proceedings*, 2004.

[35] K. HIRATA, S. MUKHERJEA, Y. OKAMURA, W.S. LI et Y. HARA. Object-based navigation: An intuitive navigation style for content-oriented integration environment. In *ACM Hypertext Proceedings*, pages 75–86, 1997.

[36] R. HJELSVOLD et R. MIDTSTRAUM. Modeling and querying video data. In *20th International Conference on very large database (VLDB)*, pages 686–694, Santiago, Chile, 1994.

[37] W. HSU, T. S. CHUA et H. K. PUNG. An integrated color-spatial approach to content-based image retrieval. In *Proceedings of the 3th ACM International Multimedia Conference (ACM-MM'95)*, pages 303–313, 1995.

[38] J. HUANG, S. R. KUMAR et R. ZABIH. An automatic classification scheme. In *Proceedings of the 6th ACM International Multimedia Conference (ACM-MM'98)*, Bristol, Royaumes-Unis, septembre 1998.

[39] R. HULL et R. KING. Semantic database modeling: Survey, applications, and research issues. *ACM Computing Surveys*, 19(3):201–260, 1987.

[40] Sun Microsystems INC. Java media framework api guide, 1999.

[41] T. ISAKOWITZ, E. A. STOHR et P. BALASUBRAMANIAN. RMM: A methodology for structured hypermedia design. *Communications of the ACM*, 38(8):34–44, août 1995.

[42] G. IYENGAR et A. LIPPMAN. Videobook: An experiment in characterization of video. In *Proceedings of the 3rd IEEE International Conference on Image Processing (ICIP)*, pages 855–858, Lausanne, Switzerland, septembre 1996.

[43] B. JAHNE. *Digital Image Processing: Concepts, Algorithms and Scientific Applications*. Springer-Verlag, 1991. 402 pages,

[44] A. K. JAIN, A. VAILAYA et X. WEI. query by video clip. In *Multimedia Systems, 7*, pages 369–384, 1999.

[45] R. JAIN et A. HAMPAPUR. Metadata in video databases. *SIGMOD record*, 23(4):27–33, 1994.

[46] H. JIANG et A. K. ELMAGARMID. Spatial and temporal content-based access to hypervideo databases. *The VLDB Journal*, 7(4):226–238, 1998.

[47] G. JOMIER, M. MANOUVRIER et M. RUKOZ. Stockage et gestion d'images par un arbre quaternaire générique. In *Acte des 15èmes Journées Bases de Données Avancées (BD1'99)*, volume octobre, pages 405–424, 1999. Bordeaux,

[48] P. M. KELLY, T. M. CANNON et D. R. HUSH. Query by image example: the candid approach. *Storage and Retrieval for Image and Video Databases III*, 2420:238–248, 1995.

[49] Y.M. KIM, S.W. CHOI et S.W. LEE. Fast scene change detection using direct feature extraction from mpeg compressed videos. In *15th International Conference on Pattern Recognition*, volume 3, pages 3–8, September 2000.

[50] S. LAWRENCE, M.F. AUCLAIR-FORTIER, D. ZIOU et A. BEGHDADI. Détection insensible au mouvement des frontières de plans franches et graduelles dans les séquences vidéo numériques. In *Actes de la conférence CORESA : Compression et Représentation des Signaux Audiovisuels*, pages 12–13, Novembre 2001.

[51] S. LEFÈVRE, J. HOLLER et N. VINCENT. Segmentation temporelle de séquences d'images en couleurs compressées et non compréssées en temps réel. In *congrès francophone ORASIS de Vision par ordinateur*, Prague, Tcheque Republic, juin 2001.

[52] S. LEFÈVRE, J. HOLLER et N. VINCENT. A review of real-time segmentation of uncompressed video sequences for content-based search and retrieval. *Real-Time Imaging*, 9, 2003.

[53] S. LEFÈVRE, C. L'ORPHELIN et N. VINCENT. Extraction multicritère de texte incrusté dans les séquences vidéo. In *Colloque International Francophone sur l'Ecrit et le Document (CIFED)*, La Rochelle, France, juin 2004.

[54] P. LEWIS, H. DAVIS, S. GRIFFITHS, W. HALL et R. WILKKINS. Content based retrieval and navigation with images in the microcosm model. In *MediaComm*, pages 86–90, Southampton, UK, avril 1995.

[55] R. W. LIENHART. Comparison of automatic shot boundary detection algorithms. In *SPIE Storage and Retrieval for Image and Video Databases VII*, volume 3656, pages 290–301, San Jose, California, janvier 1999.

[56] X. LIU, Y. ZHUANG et Y. PAN. A new approach to retrieve video by example video clip. In *ACM Multimedia (2)*, pages 41–44, 1999.

[57] X. LIU, Y. ZHUANG et Y. PAN. A new approach to retrieve video by example video clip. In *ACM Multimedia*, pages 41–44, 1999.

[58] E. LOISANT, R. SAINT-PAUL, J. MARTINEZ, G. RASCHIA et N. MOUADDIB. Browsing clusters of similar images. In *Actes des* 19^e *Journées Bases de Données Avancées (BDA)*, pages 109–128, Lyon, France, octobre 2003.

[59] R. LUTFI, M. GELGON et J. MARTINEZ. Structuring and querying documents in an audio database management system. *Multimedia Tools and Applications (MTAP)*, 24(2):105–123, novembre 2004.

[60] W. Y. MA et B. S. MANJUNATH. Netra: A toolbox for navigating large image databases. In *Proceedings of the IEEE International Conference on Image Processing (ICIP'97)*, 1997.

[61] S. MARCUS. *Multimedia database systems: issues and research directions*, chapter Querying multimedia databases in SQL, pages 263–277. Springer-Verlag, New York, 1996.

[62] S. MARCUS et V. S. SUBRAHMANIAN. Foundations of multimedia database systems. *Journal of the ACM*, 43(3):474–523, mai 1996.

[63] J. MARTINEZ et S. GUILLAUME. Colour image retrieval fitted to classical querying. *Networking and Information Systems Journal (NISJ)*, 1:251–278, 1998.

[64] J. MARTINEZ et E. LOISANT. Browsing image databases with Galois' lattices. In *Proceedings of the 17th ACM International Symposium on Applied Computing (ACM SAC), Multimedia and Visualisation Track*, pages 971–975, Madrid, Spain, mars 2002. ACM Computer Press.

[65] J. MARTINEZ et N. MOUADDIB. Multimedia and databases: a survey. *Networking and Information Systems Journal (NISJ)*, 2(1):89–123, 1999. Hermès Science,

[66] I. MBAYE, J. MARTINEZ et R. Oulad Haj THAMI. Un modèle d'indexation de la vidéo. *Ingénierie des systèmes d'information (ISI) : bases de données semi-structurées*, 8(5-6):173–196, 2003.

[67] I. MBAYE, J. MARTINEZ et R. Oulad Haj THAMI. Un modèle d'indexation pour la vidéo. In *Première Conférence Plainière STIC (CoPSTIC)*, Rabat, Maroc, 2003.

[68] I. MBAYE, J. MARTINEZ et R. Oulad Haj THAMI. Galois' lattice for video navigation in a DBMS. In *Workshop on Multimedia Content Representation, Classification and Security, MRCS'2006*, volume 4105 of Lecture Notes in Computer Science (LNCS), Springer, pages 418–425, Istanbul, Turkey, Sep 2006.

[69] I. MBAYE, R. Oulad Haj THAMI et J. MARTINEZ. A model for indexing videos and still images from the moroccan cultural heritage. In *IEEE Int. Workshop on Multimedia Signal Processing MMSP'2005*, Shangai, China, oct 2005.

[70] M. MECHKOUR. Emir2: An extended model for image representation and retrieval. In *Proceedings of the 6th International Conference on Database and Expert Systems Applications (DEXA'95)*, Londres, Royaumes-Unis, septembre 1995.

[71] C. MEGHINI. An image retrieval model based on classical logic. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 300–308, Seattle, Washington, juillet 1995.

[72] C. MEGHINI, F. RABATTI et C. THANOS. Conceptual modelling of multimedia documents. *IEEE Computer*, 24(10):23–32, octobre 1991.

[73] P. MULHEM, J. GENSEL et H. MARTIN. Modèles pour résumés adaptatifs de vidéos. *Ingénierie des systèmes d'informations*, 7(5–6):91–118, 2002.

[74] F. NACK et A. PARKES. The application of video semantics and theme representation in automated video editing. *Multimedia Tools and Application*, 4(1):57–83, janvier 1997.

[75] F. NACK et W. PUTZ. Designing annotation before it's needed. In *9th ACM International Conference on Multimedia*, pages 251–260, Ottawa, Ontario, octobre 2001.

[76] A. D. NARASIMHALU. Multimedia databases. *Multimedia Systems*, 1996.

[77] C. NASTAR, M. MITSCHKE et C. MEILHAC. Efficient query refinement for image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'98)*, Santa Barbara, Californie, juillet 1998.

[78] J. NIELSEN. *Hypertext and Hypermedia*. Academic Press, San Diego, California, 1990. 268 p.,

[79] V. E. OGLE et M. STONEBRAKER. Chabot: Retrieval from a relational database of images. *IEEE Computer*, pages 40–48, sep 1995.

[80] J. OH et K.A. HUA. Efficient and cost-effective techniques for browsing and indexing large video databases. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, pages 415–426, Dallas, Texas, USA, 16-18 mai 2000.

[81] E. OOMOTO et K. TANAKA. OVID : Design and implementation of a video-object database system. *IEEE Transactions on Knowledge and Data Engineering*, 5(4):629–643, février 1993.

[82] M. ORTEGA, Y. RUI, K. CHAKRABARTI, S. MEHROTA et T. S. HUANG. Supporting similarity queries in mars. In *Proceedings of the ACM International Conference on Multimedia (MM'97)*, 1997.

[83] G. PASS et R. ZABIH. Comparing images using joint histograms. *Multimedia Systems*, 7(3):234–240, 1999. Springer-Verlag,

[84] A. PENTLAND, R. W. PICARD et S. SCLAROFF. Photobook: Content-based manipulation of image databases. *International Journal of Computer Vision*, 18(3):233–254, 1995.

[85] M. PETKOVIĆ et W. JONKER. A framework for video modelling. In *18th IASTED Conference on Applied Informatics*, Innsbruck, Austria, février 2000.

[86] M. PETKOVIĆ, W. JONKER et V. MIHAJLOVIC. Extending a DBMS to support content-based video retrieval: A formula 1 case study. In *2nd International Workshop on Multimedia Data Document Engineering (MDDE)*, volume LNCS 2490, pages 318–341, Prague, Czech Republic, mars 2002. Springer-Verlag, Heidelberg.

[87] R. W. PICARD. A society of models for video and image libraries. *IBM Systems Journal*, 35(3-4), 1996.

[88] R. W. PICARD et T. P. MINKA. Vision texture for annotation. *Multimedia Systems*, 3(1):3–14, février 1995.

[89] Y. PRIÉ, A. MILLE et J. M. PINON. Une approche de modélisation de documents audiovisuels en strates interconnectées par les annotations. In *Ingénierie des Connaissances (IC)*, pages 143–152, Pont-à-Mousson, France, mai 1998.

[90] U. PRISS. Lattice-based information retrieval. *Knowledge Organization*, 27(3):132–142, 2000.

[91] Y. PRIÉ, A. MILLE et J. M. PINON. Ai-strata : a user-centered model for content-based description and retrieval of audiovisual sequences. In *International Advanced Multimedia Content Processing Conference*, novembre 1998.

[92] A. Hanjalic R., R. LGENDIJK et J. BEIMOND. Automated high-level movie segmentation for advanced video retrieval systems. *IEEE Trans. on Circuits and Systems for Video Technology*, 9(4):580–588, 1999.

[93] M. RAUTIAINEN, T. OJALA et T. Seppänen AND. Cluster-temporal browsing of large news video databases. In *IEEE International Conference on Multimedia and Expo ICME*, pages 751–754, 2004.

[94] H. REHATSCHEK et G. KIENAST. Vizard - an innovative tool for video navigation, retrieval, annotation and editing. In *Proceedings of the 23rd Workshop of PVA : Multimedia and Middleware*, Vienna, mai 2001.

[95] Y. RUI, T. S. HUANG et S. MEHROTRA. Exploring video structure beyond the shots. In *IEEE International Conference on Multimedia Computing and Systems (ICMCS)*, pages 237–240, Austin, Texas, juin 1998.

[96] E. SABER, A. M. TEKALP, R. ESCHBACH et K. KNOX. Automatic image annotation using adaptative color classification. *Graphical Model and Image Processing*, 58(2):115–126, mars 1996. Academic Press,

[97] R. SAINT-PAUL, G. RASCHIA et N. MOUADDIB. Prototyping and browsing image databases using linguistic summaries. In *Proc. of the IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE'2002)*, Honolulu (Hawaii), USA, May 2002.

[98] H. SAMET. The quadtree and related hierarchical data structures. *ACM Computing Surveys*, 16(2):188–260, juin 1984.

[99] S. SANTINI et R. JAIN. Similarity is a geometer. *Multimedia Tools and Applications*, 5:377–406, nov 1997.

[100] S. SANTINI et R. JAIN. Beyond query by example. In *Proceedings of the 6th ACM International Multimedia Conference (ACM-MM'98)*, Bristol, Royaumes-Unis, septembre 1998.

[101] D. SCHWABE, G. ROSSI et S. D. J. BARBOSA. Systematic hypermedia application design with OOHDM. In *Proceedings of the 7th ACM Conference on Hypertext*, pages 116–128, Washington, D. C., 16-20 mars 1996.

[102] M. K. SHAN et S. Y. LEE. Content-based video retrieval based on similarity of frame sequence. In *IEEE Proc of Int'l Workshop on Multimedia Database Management Systems*, 1998.

[103] A. SMITH, G. THOMAS et G. DAVENPORT. The stratification system: Design environnement for random access video. In 3^{rd} *ACM International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, volume 712 of *Lecture Notes in Computer Science*, pages 250–261, La Jolla, California, novembre 1992. Springer.

[104] J. R. SMITH et S. F. CHANG. Visualseek: A fully automated content-based image retrieval system. In *Proceedings of the 4th ACM International Multimedia Conference (ACM-MM'96)*, Boston, Massachusets, nov 96.

[105] A. STEFANIDIS, P. PARTSINEVELOS, K. EICKHORST et P. AGOURIS. Spatiotemporal lifelines in support of video queries. In *12th IEEE International Workshop on Database and Expert Systems Applications (DEXA)*, pages 865–869, Munich, Germany, septembre 2001.

[106] M. STRICKER et A. DIMAI. Color indexing with weak spatial constraints. In *Storage and Retrieval for Image and Video Databases IV, SPIE Proceedings Series*, volume 2670, pages 29–40, février 1996.

[107] M. STRICKER et M. ORENGO. Similarity of color images. In *Storage and Retrieval for Image and Video Databases III, SPIE Proceedings Series*, volume 2420, pages 381–392, 1995.

[108] M. J. SWAIN et D. H. BALLARD. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[109] M. SZUMMER et R. W. PICARD. Indoor-outdoor image classification. *Proceedings of the International Workshop on Content-Based Access of Image and Video Databases (CAIVD)*, 1998. IEEE Computer Society,

[110] H. TAMURA, S. MORI et T. YAMAWAKI. Textural features corresponding to visual perception. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-8(6), 1978.

[111] Y. P. TAN, S. R. KULKARNI et P. J. RAMADGE. A framework for measureing video similarity and its application to video query by example. In *IEEE Int'l COnf on Image Processing*, October 1999.

[112] Y. TAO et I. GROSKY. Spatial color indexing: A novel approach for content-based image retrieval. In *Proceedings of the IEEE International Conference on Multimedia Computing and Systems (ICMCS'99)*, pages 530–535, Florence, Italy, juin 1999.

[113] R. Oulad Haj THAMI, T. Filali ANSARY, M. DAOUDI et H. CHAARANI. Generic: un système pour la rehceche d'images basée sur les attributs visuels et les connaissances. In *Compression et Représentation des Signaux Audiovisuels (CORESA'2003)*, 16-17 janvier 2003.

[114] T. THUONG. Description de la structure des vidéos pour les applications multimédias. Master's thesis, Imagerie Vision et Robotique, Grenoble, France, 2001.

[115] R. TUSCH, H. KOSCH et L. BÖSZÖRMENYI. VideX: An integrated generic video indexing approach. In *ACM International Multimedia Conference*, Los angeles, California, octobre 2000.

[116] E. VACHON. A pre-viewing step in video retrieval. In *International Workshop on Multimedia Data and Document Engineering (MDDE)*, Prague, Czech Republic, mars 2002.

[117] E. VACHON et A. DOUCET. Armitage, un entrepôt virtuel de vidéos orienté vers la validation de requêtes vidéos. *Ingénierie des systèmes d'informations*, 7(5–6):141–168, 2002.

[118] A. VAILAYA, A. JAIN et H.-J. ZHANG. On image classification: City image vs. landscapes. *Pattern Recognition*, 31(12):1921–1936, 1998. Hermès Science,

[119] P. VALTCHEV, R. MISSAOUI et P. LEBRUN. A fast algorithm for building the hasse diagram of a galois lattice. A paraître dans Discrete Mathematics, 2001,

[120] N. VASCONCELOS et A. LIPPMAN. Statistical models of video structure for content analysis and characterization. *IEEE Trans. on Image Pro-cessing*, 9(1):3–19, 2000.

[121] M. VAZIRGIANNIS. Uncertainty handling in spatial relationships. In *Proceedings of the 2000 ACM Symposium on Applied Computing (SAC'00)*, volume 1, pages 494–500, mars 2000.

[122] E. VENEAU. *Macro-segmentation multi-critère et classification de séquences par le contenu dynamique pour l'indexation vidéo*. Thèse de Doctorat, Université de Rennes, France, 2002.

[123] E. VENEAU, R. RONFARD et P. BOUTHEMY. From video shot clustering to sequence segmentation. In 15^{th} *IEEE International Conference on Pattern Recognition (ICPR)*, Barcelona, Spain, septembre 2000.

[124] H. WACTLAR, M. CHRISTEL, Y. GONG et A. HAUPTMANN. Lessons learned from the creation and deployment of a terabyte digital video library. *IEEE Computer*, 32(2):66 – 73, 1999.

[125] H. WANG, F. GUO et D. D. FENG. A signature for content-based image retrieval using a geometrical transform. In *Proceedings of the 6th ACM International Multimedia Conference (ACM-MM'98)*, Bristol, Royaumes-Unis, septembre 1998.

[126] R. WILLE. Restructuring lattice theory: an approach based on hierarchies of concepts. In Ivan RIVAL, réd., *Ordered Sets*, pages 445–470, Dordrecht-Boston, 1982. Reidel.

[127] C. WOLF et J. M. JOLION. Extraction de texte dans des vidéos : le cas de la binarisation. In *Proceedings of RFIA 2002*, volume 1, pages 145–152, January 2000.

[128] M. E. J. WOOD, N. W. CAMPBELL et B. T. THOMAS. Iterative refinement by relevance feedback in content-based digital image retrieval. In *Proceedings of the 6th ACM International Multimedia Conference (ACM-MM'98)*, Bristol, Royaumes-Unis, septembre 1998.

[129] A. WOUDSTRA, D. D. VELTHAUSZ, H. J. G. DE POOT, F. MOELAERT EL-HADIDY, Willem JONKER, M. A. W. HOUTSMA, R. G. HELLER et J. N. H. HEEMSKERK. Modeling and retrieving audiovisual information — A soccer video retrieval system.

In *4th International Workshop on Multimedia Information Systems (MIS)*, volume 1508 of *Lecture Notes in Computer Science*, Istanbul, Turkey, septembre 1998.

[130] J. K. WU, A. D. NARASIMHALU, B. M. MEHTRE, C. P. LAM et Y. J. GAO. Core: A content-based retrieval engine for multimedia information systems. *Multimedia Systems*, 3(1):25–41, février 1995.

[131] E. ZENOU et M. SAMUELIDES. Utilisation des treillis de galois pour la caractérisation d'ensembles d'images. In *Reconnaissance de Formes et Intelligence Artificielle (RFIA)*, 2004.

[132] H. J. ZHANG, Y. GONG, S. W. SMOLIAR et S. Y. TAN. Automatic parsing of news video. In *International Conference on Multimedia Computing and Systems (ICMCS)*, pages 45–54, Boston, Massachusetts, USA, mai 1994.

Annexes

ANNEXE A

LE LANGAGE CINÉMATOGRAPHIQUE

Le cinéma est un langage d'expression qui s'est très rapidement répandu. Contrairement au langage verbal qui utilise une seule expression phonique et plusieurs codes, le langage cinématographique combine cinq matières d'expression (présentes dans la bande image et sonore) et plusieurs codes. La bande image est constituée des images animées ainsi que des représentations graphiques contenues dans chaque image ; la bande son comporte paroles, musique et bruits.

Avec le développement des technologies numériques, on a assisté à la numérisation progressive de l'ensemble de la chaîne de production d'un film. La fabrication d'un film suppose désormais de nombreux aller-retours entre analogique et numérique.

En fait, la numérisation du cinéma ne se déploie réellement à l'échelle industrielle qu'à partir des années 1980. La vidéo est la numérisation complète des séquences filmées. À la fin de cette décennie, la numérisation touche aussi les télévisions et bouleverse les modes d'exploitations des archives audiovisuelles. Désormais, il est possible d'archiver les données audiovisuelles sur des supports numériques (disques durs, cédéroms, etc.) afin qu'elles puissent être reconnues, traitées et diffusées.

Notre travail a porté sur l'analyse de la bande image d'une vidéo afin de la décrire pour pouvoir la rechercher par la suite. Pour assurer une compréhension complète des messages qu'elle véhicule, il est donc important de l'analyser afin de comprendre sa grammaire et ses règles de montage. Avant cette analyse, il est important de lever certaines confusions :

1. Problème de la définition de l'unité significative minimale : Pour certains chercheurs, c'est le plan qui constitue l'unité significative minimale [8] ; pour d'autres c'est l'image [17], et pour d'autres encore ce sont les objets contenus dans les images [13]. Cette

confusion ne résulte que d'un malentendu car la multitude de codes présents dans la bande d'images permet le choix de n'importe laquelle de ces informations comme unité minimale.

2. Problème lié à l'analyse du sens des plans : Très souvent, le cinéma ne représente pas directement la réalité mais a recours à des techniques de présentations des événements pas toujours faciles à comprendre. Par exemple, le plan d'un visage associé à l'image d'une assiette de soupe peut signifier « faim ».

Dans le cadre de l'interrogation des vidéos extraites de films, la sémantique des techniques cinématographiques est donc à prendre en compte. Aussi, donnons-nous un aperçu sur le processus allant de l'écriture d'une histoire jusqu'à sa traduction en images animées, car certaines recherches tentent de modéliser le plus exhaustivement possible ces techniques-là [23]. Ces modélisations tentent de dissocier les différentes composantes d'un film (caméra, histoire, personnage, etc.).

A.1 Réalisation d'un film

La réalisation d'un film suit généralement cinq étapes principales :

- écriture de l'histoire ;
- écriture du scénario ;
- découpage technique ;
- prises de vue ;
- montage.

A.1.1 Écritures de l'histoire et du scénario

L'écriture de l'histoire consiste à rédiger quelques lignes à quelques pages exposant l'idée générale du film. Le scénario précise scène par scène le déroulement et les péripéties de histoire. Il fixe les décors, les actions, les dialogues des personnages etc. L'état final du scénario est le *storyboard* qui est une suite de dessins correspondant chacun à un plan. Dans le *storyboard*, on précise le type du plan et du mouvement de caméra qui sera utilisé pour chaque plan ainsi qu'une description des actions dans le plan.

Définition A.1 (Plan)**.** Le plan est une suite d'images filmées sans interruption par la caméra.

Définition A.2 (Action)**.** Une action est un ensemble d'événements d'un récit, d'un drame, considérés dans leur progression : intrigue, mouvement, rythme de cette intrigue.

A.1.2 Découpage technique

Le découpage du scénario se fait en fonction des plans à tourner. Il enrichit le scénario des indications techniques nécessaires au tournage du film. On y introduit une énumération des plans ainsi que des indications qui s'y rapportent telles que les indications techniques au sujet des cadrages, des mouvements d'appareils, des effets visuels et sonores devant apparaître dans le film, des angles de prise de vue, des échelles des plans etc. Les films ne se tournent pratiquement jamais dans l'ordre dans lequel ils sont regardés. On filme généralement à la suite les plans se déroulant dans un même lieu et nécessitant les mêmes acteurs. En un mot, le découpage technique précise l'ordre dans lequel le film sera tourné.

A.1.3 Prises de vue

Lorsque le scénario est prêt, découpé et comporte toutes les indications techniques nécessaires, on passe au tournage du film. C'est l'étape de prises de vue. Une prise de vue correspond à la suite d'images comprises entre le déclenchement de la caméra et son arrêt (c'est-à-dire un plan). Pour certains types de film comme les documentaires, les reportages, etc., la réalisation du film commence directement avec la prise de vue. L'effet d'une prise de vue dépend énormement de l'angle de prise de vue et des échelles des plans. Ces deux facteurs vont fortement influencer sur la perception de l'image finale.

Précisons ces deux facteurs :

- Angle de prise de vue : Cet angle détermine la position dont sera disposée la caméra pour filmer une scène. Cette position permet de distinguer entre :
  - Une prise de vue horizontale : La caméra est située au même niveau de ce qu'elle filme. Cette position donne un cadre parfaitement horizontal.
  - Plan cassé : Dans ce cas, la caméra n'est pas tout à fait horizontale. Le contenu de l'image semble pencher d'un côté ou de l'autre.
  - Une plongée : La plongée consiste à placer la caméra au dessus de la scène qu'elle filme. L'effet produit est l'écrasement d'un personnage. L'image qui en découle est donc légèrement déformée et le regard des personnes n'est pas clairement perceptible.
  - Une contre-plongée : Inversement, la contre-plongée consiste à placer la caméra en dessous de la scène qu'elle filme. À l'inverse de la plongée, les personnages paraîtront plus grands sur l'image.
- Échelle des plans : Ce second facteur important de la prise de vue représente le rapport entre le cadre et les objets contenus dans le cadre. C'est en fait la grandeur

des êtres ou des objets de l'espace représentés dans l'image par rapport à la taille de l'image. Il existe deux types d'échelles qui engendrent chacun plusieurs types de plans différents :

- L'échelle par rapport au décor comprend :
  - Le plan général : Un plan général cadre l'ensemble d'un décor, d'un paysage.
  - Le plan d'ensemble : Le plan d'ensemble précise le décor. Il tend à créer une synthèse, un cadre descriptif du décor.
  - Le plan moyen : Un plan moyen cadre un ou plusieurs personnages en pied. Il concentre l'attention du spectateur sur un personnage particulier.
- L'échelle par rapport au personnage comprend :
  - Le plan italien : Le plan italien est encore appelé plan genou car il présente les personnages jusqu'au genou. Il rapproche le spectateur des personnages.
  - Le plan américain : Le plan américain, ou plan cuisses, présente les personnages jusqu'au cuisses. Il rapproche plus encore le spectateur des personnages.
  - Le plan rapproché : Le plan rapproché coupe les personnages soit à la taille, soit à la poitrine par rapprochement de la caméra.
  - Le gros plan : Le gros plan ne présente que la tête d'un personnage.

En plus de l'angle de prise de vue et des échelles de plan, il arrive souvent qu'on effectue une prise de vue dans laquelle le décor ou le personnage change légèrement sans changement de plan. Dans de telles situations, on effectue un mouvement de caméra pour suivre l'action sans arrêt de la caméra : c'est le mouvement de caméra.

Définition A.3 (Mouvements de caméra)**.** Un mouvement de caméra est un déplacement de la caméra visant à donner l'impression de mouvement dans des images filmées ou enregistrées ou à en modifier la perspective.

Il existe plusieurs types de mouvement de caméra :

- Le panoramique : Un panoramique (horizontal, vertical, circulaire) est réalisé lorsque la caméra fixée au sol pivote sur son axe. Le panoramique permet de découvrir un panorama. Selon les contextes, les utilisations seront multiples : suivre un personnage, découvrir un nouveau lieu, etc.
- Le travelling : Le travelling (avant, arrière, latéral ou vertical) se produit quand la caméra se déplace par rapport à la scène qu'elle filme. On parle de travelling avant quand elle s'en approche et de travelling arrière quand elle s'en éloigne. Un travelling allant vers un objet ou un personnage le met en valeur. À l'inverse un travelling arrière est une sorte de conclusion : on se retire de l'action, de l'histoire. Le travelling peut être aussi latéral ; il suit le déplacement d'un personnage, horizontalement ou

verticalement (en montée ou en descente).

- Le zoom : Le zoom (ou travelling optique) permet de rapprocher ou d'éloigner la scène du spectateur sans que la caméra ne se déplace.

Les données acquises par les caméras sont uniquement des informations en deux dimensions puisque la formation des images résulte de la projection de la scène observée sur un plan. Par conséquent, et contrairement au système de vision humaine, il n'y a aucune connaissance directe de la structure tridimensionnelle de la scène. On parlera ainsi de mouvements *apparents* dans des séquences d'images, résultants de la projection de mouvements en trois dimensions.

Enfin, les scènes filmées, peuvent être plus ou moins éclairées.

Définition A.4 (Éclairage)**.** L'éclairage est le jeu des ombres et des contrastes appliqué aux éléments filmés. L'éclairage peut être fait par devant, par derrière ou encore mixte (devant-derrière). Chaque type d'éclairage à une incidence précise sur la scène filmée.

A.1.4 Montage

Comme nous l'avons précisé, les films ne se tournent pratiquement jamais dans l'ordre dans lequel ils seront regardés. On filme généralement à la suite les plans se déroulant dans un même lieu et nécessitant les mêmes acteurs. Ensuite il faut recoller bout-à-bout les plans afin de reconstituer le scénario : c'est l'étape de montage du film. Le montage est l'art d'exprimer ou de signifier par le rapport de deux plans juxtaposés de telle sorte que cette juxtaposition fasse naître l'idée ou exprime une émotion qui n'est contenue dans aucun des plans pris séparement. L'ensemble est supérieur à la somme des parties. Les plans peuvent être juxtaposé de différentes manières pour former des unités de granularité plus grande.

Ces juxtapositions peuvent se faire de différentes façons *via* des transitions.

Définition A.5 (Transition)**.** Une transition est le passage d'un plan à un autre. Il existe plusieurs types de transitions :

- La coupe : La coupe, plus fréquemment dénommée *cut*, consiste à un passage brusque d'un plan à un autre.
- Le fondu : Un fondu (*fade*) correspond à l'apparition ou à la disparition progressive d'une image. Il existe plusieurs formes de fondus :
  - Le fondu enchaîné (*cross fading*): Il désigne le fait de passer progressivement d'un plan à un autre par un effet qui est le remplacement progressif donc la superposition de deux images dans l'étape de fondu.

  - Le fondu au noir : Encore appelé fondu fermé ou fermeture en fondu *(dissolve, fade out)*, il désigne l'assombrissement progressif d'une image jusqu'à sa disparition en une image entièrement noire.
  - Le fondu au blanc : Inversement, il désigne l'éclaircissement progressif d'une image jusqu'à sa disparition en une image entièrement blanche.
  - Le fondu ouvert ou ouverture en fondu *(fade in)* : Il s'agit de l'apparition progressive d'une image qui semble émerger du noir.
- Le volet : Le volet (*wipe*) consiste à découvrir une image par une sorte de rideau.

La juxtaposition des plans permet d'avoir des unités de granularité plus grandes comme les scènes et les séquences.

Définition A.6 (Scène)**.** Une scène est composée de plans filmés en un même moment et au même endroit.

Définition A.7 (Séquence)**.** La séquence est une suite de scènes dans laquelle l'unité de l'action est conservée.

Des effets spéciaux comme les *inserts*, les ellipses, les plans de coupe, etc., peuvent aussi être introduits pour raccorder les plans :

- L'ellipse permet de supprimer des passages moins significatifs pour se concentrer sur ce qui constituent l'essentiel d'un récit.
- L'*insert* est un plan de courte durée qu'on introduit dans un montage de manière à mettre en évidence un détail de la scène, un élément particulier ou un geste d'un personnage.
- Le plan de coupe est un plan de courte durée et sans signification majeure inséré au montage entre deux plans qui ne se raccordent pas parfaitement.
- Le plan-séquence est filmé d'une seule traite et restitué tel quel dans le film final, c'est-à-dire sans montage ou interruption de point de vue (sans plan de coupe, fondu, volet ni contre-champ). Il a une unité sur le plan narratif (c'est une séquence) et sur le plan technique (c'est un plan), d'où son nom.

A.2 Média vidéo

Rappelons que la vidéo est la numérisation complète d'une séquence d'images filmées. De nos jours, on assiste à une convergence entre télévision et ordinateur qui est en train de changer notre manière d'utiliser ces deux types d'appareils. Cependant, la vidéo présente

de nombreux problèmes pour l'intégrer dans des systèmes informatiques en général et pour construire des SGBD multimédia ou des serveurs vidéos en particulier.

Le premier problème est lié à la nature même d'une vidéo : c'est une séquence d'images liées à une bande son (souvent à une ou plusieurs pistes texte) synchronisée. Sa sémantique passe donc par celle des images, de la bande son mais aussi par celle de leur combinaison. Il est donc important de prendre en compte la dimension spatio-temporelle des vidéos.

Le deuxième problème vient de la taille de ce type de fichier. Un film d'une heure peut atteindre des gigaoctets. Les algorithmes de compression / decompression (CODEC) permettent de réduire la taille de ces fichiers par condensation de données répétitives contenues dans un film. Chaque algorithme de compression génère un format de vidéo spécifique.

A.2.1 Contenu

A.2.1.1 Images animées

Tous les documents (ou flux) vidéo contiennent une piste « image ». Les éléments de cette piste sont des images émises à une fréquence fixe (typiquement 24 à 30 images par secondes). Certains documents vidéos peuvent contenir plusieurs pistes d'images en parallèle ; ce cas est toutefois assez rare.

A.2.1.2 Son

La plupart des documents (ou flux) vidéo contiennent aussi une ou plusieurs pistes « audio ». Les éléments de cette piste sont des échantillons émis à une fréquence fixe (typiquement de 16 000 à 48 000 par secondes). Une piste audio peut encore être composée de plusieurs flux de tels éléments en parallèle (c'est le cas de la stéréo). Un document (ou flux) vidéo peut contenir plusieurs pistes audio en parallèle (correspondant à plusieurs langues par exemple).

A.2.1.3 Texte

Certains documents (ou flux) vidéo contiennent aussi une ou plusieurs pistes textuelles. Les éléments de cette piste ne sont généralement pas émis à une fréquence fixe mais plutôt par paquets accompagnés des informations permettant de les synchroniser avec les autres flux (temps de début et de fin), alors que dans le cas des pistes d'images et de son, la synchronisation se fait sur la base de l'émission régulière et à une fréquence fixe des éléments de base (image et échantillon sonore).

Standard	Taille de l'image	Octets/pixel	Images/s	Mo/s
NTSC	640×480	3	30	27,6
PAL	768×576	3	25	33,2
SECAM	625×468	3	25	22,0
CCIR	720×486	2		21,0

Table A.1 – Standards pour la vidéo

A.2.2 Normes

Le tableau A.1 présente différents standards de vidéo. Le standard NTSC (*National Television System Committee*) est en vigueur aux États-Unis et au Japon principalement. Le standard PAL (*Phase Alternating Line*) est en vigueur en Allemagne, en Grande-Bretagne et en Chine, entre autres. Le système SECAM (SÉquentiel Couleur À Mémoire) est utilisé en France et en Russie. Le CCIR601 mis en place par le CCIR (Comité Consultatif International de Radio) pour la télévision digitale est à la base de différents formats d'échanges.

Tous ces formats correspondent à de la vidéo « brute », non compressée. Plusieurs propositions ont été faites pour réduire le volume de la vidéo que l'on stocke.

A.2.3 Compression

La compression de fichiers offre des intérêts majeurs. Elle facilite le stockage, l'accès rapide aux données stockées, le transfert sur le réseau. Les codecs offrent deux principales manières d'évacuer les données répétitives : soit par compression spatiale, soit par compression temporelle.

A.2.3.1 Compression spatiale

Dans une image vidéo, il peut y avoir plusieurs zones dont les pixels sont exactement de la même couleur. Le codec, au lieu de donner la position de chaque pixel et sa couleur, généralise le tout en spécifiant plutôt les coordonnées de la zone ainsi que la couleur de cette zone. Cette manière de réduire la taille d'une image se nomme la compression spatiale. Moins il y a de détails dans une image, plus la compression spatiale sera importante.

A.2.3.2 Compression temporelle

Une autre manière d'évacuer des données consiste à analyser les changements entre plusieurs images consécutives et à ne conserver que ceux-ci plutôt que les images complètes.

On choisit une image de référence pour le début de l'analyse. La description de l'image de référence est complète, tandis que les images suivantes ne sont pas stockées complètement. On conserve seulement une description des différences avec l'image précédente, ce qui, habituellement, prend bien moins de place que la description complète des images. On déclare de nouvelles images de référence à différents intervalles pour éviter que la somme des changements entre les images n'introduise des erreurs cumulatives et ainsi une dégradation trop importante de la qualité par rapport aux images originales. Cette méthode se nomme la compression temporelle. Les séquences qui comportent peu de mouvements sont les meilleurs sujets pour la compression temporelle.

A.2.4 Algorithmes de compression

On distingue deux grandes catégories d'algorithmes de compression :

- avec pertes ;
- sans perte.

Les algorithmes de compression avec pertes sont les plus courants. Cela signifie que le résultat décompressé n'est pas parfaitement identique à l'information originale. Mais la différence entre les versions originales et les versions compressées puis décompressées sont souvent peu perceptibles par un humain grâce à l'exploitation des redondances spatiales et temporelles présentes dans les documents et grâce à l'exploitation des caractéristiques des systèmes perceptifs humains par les algorithmes de compression et de décompression (ce que nous ne percevons pas peut être supprimé ou rendu de manière approximative).

Ces algorithmes jouent donc sur le fait que, dans les données vidéo, tout n'est pas perceptible de la même manière par l'être humain. On sait par exemple que l'œil est plus sensible à des changements, même faibles, dans la brillance que dans la couleur. Il suffit donc d'allouer dans la compression plus de bits au codage de la luminance qu'à celui de la chrononinance. Bien sûr un compromis doit être trouvé entre le facteur de compression et la qualité de la vidéo. Plus la compression est forte, plus la taille des données se réduit mais au détriment de la qualité. Inversement, si l'on diminue le facteur de compression, la taille des données augmente mais la qualité est meilleure.

A.2.5 Structure de la vidéo

Les documents vidéo peuvent avoir des contenus extrêmement variés. Ces contenus dépendent du type de la vidéo (journaux télévisés, documentaires, films, publicité, etc.). La plupart du temps, ces documents ont une ou plusieurs structures internes. Comme les docu-

ments eux-mêmes, ces structures peuvent être très variées. Les structures dont nous parlons ici sont conceptuellement différentes des structures physiques dont nous avons parlé dans la section précédente bien qu'il arrive souvent qu'elles coïncident en pratique lorsque les deux types existent. Le type de structure dont il est question ici est relatif au contenu sémantique du document. Il s'agit de structures qui ont un sens pour l'utilisateur. À ce titre, elles peuvent parfois apparaître mal définies, ambiguës ou subjectives. Ces structures se présentent souvent de manière hiérarchique : un document est décomposé hiérarchiquement en unités plus petites selon un arbre (pas forcément régulier, et notamment parfois de profondeur variable).

A.3 Conclusion

Dans cette annexe, nous avons introduit les concepts de base du langage cinématographique, l'objectif étant de fournir un minimum de culture cinématographique au lecteur qui n'aurait pas encore prêté attention à la façon dont les documents audiovisuels sont assemblés. De la sorte, nous avons introduit et spécifié le vocabulaire cinématographique associé à un certain nombre de notions que nous utilisons dans ce document : documents vidéo, plan, séquence, etc. Nous avons également évoqué quelques spécificités des documents vidéo notamment les aspects de multimodalité et d'hétérogénéité du contenu, même si la multimodalité de la vidéo n'est pas traitée dans cette thèse.

Résumé

Dans cette thèse, nous avons développé $Find^{Me}_{DIA}$ destiné à répondre aux besoins de conservation du patrimoine culturel marocain filmé et photographié.

Nous avons donc à gérer une base d'images et de vidéos. Une vidéo pouvant être perçue comme une succession d'images fixes, nous avons traité les vidéos comme une extension qui s'appuie sur la modélisation des images de manière quasi transparente. Notre principal but est, d'une part, de répondre aux besoins de *généricité* et de *flexibilité* permettant de traiter ces différents types de médias visuels et, d'autre part, de proposer un système qui permette de naviguer en basculant indistinctement entre images et vidéos.

Pour la modélisation des vidéos, nous avons proposé $Find^{Vi}_{DEO}$ [66]. En partie modèle, en partie métamodèle, $Find^{Vi}_{DEO}$ est flexible et englobe une large gamme d'applications et de modèles préexistants.

Pour la navigation, nous appliquons la technique des treillis de Galois sur une base de données composée d'images clés extraites des vidéos ainsi que d'images fixes. Le système $Find^{Me}_{DIA}$ [68, 69] résultant est générique et offre la possibilité d'utiliser plusieurs techniques de descriptions des images en vue de la navigation.

Pour tester l'intérêt de nos approches, la modélisation des images clés (extraites des vidéos) et des images fixes est effectuée par $Click^{Im}_{AGE}$ [64] qui propose une représentation semi-structurée des données basée sur le contenu des images.

Mots-clés : vidéo, image, indexation, modélisation, navigation, treillis de Galois

Abstract

In this work, we developed $Find^{Me}_{DIA}$, the purpose of which is to preserve the Moroccan cultural heritage in the form of movies and photographs.

Therefore, we has to manage an joint image and video database. Since a video can be perceived as a sequence of fixed images, we have been treating a video as an extension which depends on the modelling of images in a quasi-transparent way. Our main objective was, on the one hand, to meet the genericity and flexibility needs allowing to navigate with different types of visual data, and, on the other hand, to put forward a system which enables us to navigate by moving indistinctly between images and videos.

As far as the modelling of video is concerned, we proposed $Find^{Vi}_{DEO}$ [66]. Both in its model and metamodel parts, $Find^{Vi}_{DEO}$ is flexible and includes a large spectrum of applications and pre-existing models.

For the sake of navigation, we reused a Galois' lattice technique on a database composed of still images and key-frames extracted from videos. The resulting $Find^{Me}_{DIA}$ system [68, 69] is generic and enables us to use many image description techniques for navigation.

To test the interest of these approaches, the modelling of key-frames (extracted from videos) as well as still images is carried out by $Click^{Im}_{AGE}$ [64] which proposes a semi-structured representation of data based on the content of images.

Keywords: video, image, indexation, modelling, navigation, Galois' lattice